

# Automatic Model Generation for PCB-based Power Electronics

Bahaeddine Ben Hamed, Guillaume Regnat, Guillaume Lefèvre

Mitsubishi Electric R&D Centre Europe

1 Allée de Beaulieu

35 708 Rennes, France

Cyril Buttay

Université de Lyon,

CNRS, INSA-Lyon, École Centrale de Lyon,

Université Claude Bernard Lyon 1, Ampère, F-69621

Email: cyril.buttag@insa-lyon.fr

**Abstract**—This paper describes a workflow which allows to automatically go from the printed circuit board design of a power converter to a circuit model which includes circuit parasitics (a complete "virtual prototype"). Commercial software (Altium, Ansys Q3D, LTSpice) is used, with custom code at the interfaces. Some systematic simplifications are proposed to accelerate computations. This workflow is demonstrated on an example case and compared to experimental measurements (double pulse waveforms and impedance measurements).

## I. INTRODUCTION

Designing a new power electronic converter still largely relies on expertise and hardware prototyping [1], despite a continuous drop in the cost of computing power and the availability of an increasingly large range of modelling and simulation software. This can be explained by a variety of reasons, such as the presence of strong non-linearity and multi-physics couplings, or by the large variations in time scales (from nanoseconds for switching events up to minutes or hours to reach thermal steady state). As a consequence, many research projects have focused on generating models with a reduced complexity [2], [3], on implementing efficient modelling techniques (such as the Partial Element Equivalent Circuit –PEEC– for the calculation of stray inductances, in Cedrat's InCa3D or FastFieldSolver's Fasthenry), or even on exploring high performance computing platforms (such as GPUs [4]).

One additional roadblock which prevents power electronic designers from moving to "virtual prototypes" is the lack of a unified system description: power electronics systems are usually a heterogeneous assembly of packaged devices, mechanical parts, Printed Circuit Boards (PCB)... Such complex arrangement is not fully described in a single place (mechanical software for the geometry, circuit simulator for the components, etc.). As a result, preparing a model for computation can require a lot of manual efforts, or to exclude part of the converter [1]: a typical example is the extraction of the circuit parasitics of a power module, which often excludes any coupling with the gate driver PCB, despite both part being very close from each other. Furthermore, packaged devices are "black boxes", with all their internal details (layout, chip distribution, etc.) hidden away from the circuit designer. As a consequence, they cannot be modeled in the same way

as the rest of the converter. The designer must rely on the manufacturer's devices models, if any.

Because of their low thermal performance and limited current capability, PCBs have long been limited to low power conversion (typically 10 or 100s of watts, for consumer applications). Recent improvements such as the PCB embedding technology [5], in which power devices can be inserted within a PCB, or the availability of thick copper layers have made PCBs much more attractive for converters in the multi-kilowatt range (3.3 kW in [6], or 50 kW in [7]). As a result, a full converter (including power, control, etc.) can be built using only PCB for its interconnects, with bare-dies power semiconductor devices. One consequence of this "rationalized" assembly is that all information about the converter becomes available in the PCB design software [8]: complete description of the layout, bill of materials (list of components), etc. In theory, it then becomes possible to use this information to generate models (thermal, electrical, etc.) in an automated way.

In practice, generating models from PCB design software is anything but trivial: in addition to the complexity issues described above, model preparation requires a lot of user interaction. Recently, Hoffmann *et al.* [9] presented a solution which allows a user to select conductors in a PCB and automatically calculate parasitic inductance, resistance, and joule heating; the objective of that paper is to provide "immediate value quantification", thanks to fast computation algorithms and to the reduction of the domain to a few conductors only. In contrast, the approach we present here aims at generating models for the entire PCB (at the cost of longer computing times). Once a PCB design is complete, the parasitic elements (capacitance, resistance, inductance and their couplings) of each track are calculated, and inserted in a circuit model along with all the components of the PCB, so as to constitute a complete "virtual prototype" of the circuit.

## II. MODEL GENERATION

The modeling approach presented here relies on existing pieces of software: electronic Computer-Aided Design (eCAD), Finite Elements (FE)/Method of Moments (MoM)/circuit simulators. These tools are mature and can manage a high degree of complexity in the models. Our main contribution consist in interfacing them so as to allow the automatic generation of a circuit model for a complete PCB.

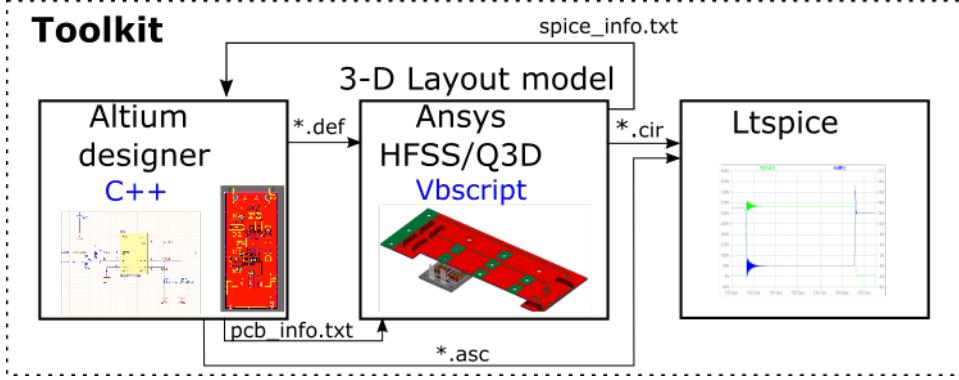


Fig. 1: The workflow for model generation

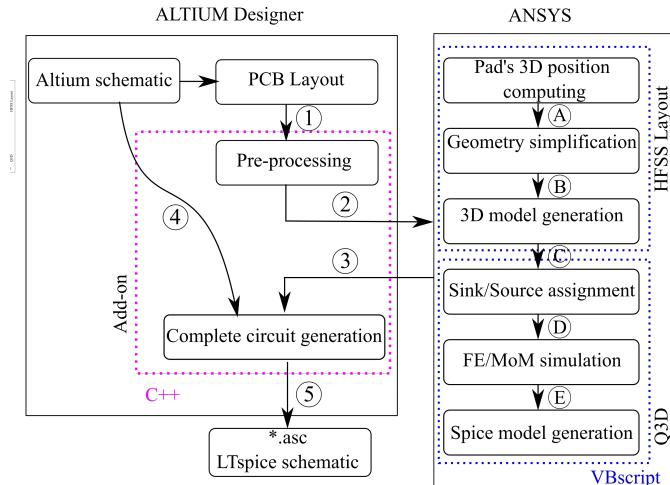


Fig. 2: Description of the toolkit workflow

The workflow used to generate circuit models from the eCAD is depicted in Fig. 1. In this paper, the following pieces of software are used: Altium Designer (eCAD), Ansys suite (and in particular Q3D Extractor to compute circuit parasitics), LTSpice for circuit simulation. While most of the data import/export relies on existing file formats, these do not capture all the information required (this is described below). As a workaround, a custom Altium extension, written in C++, generates an additional file (`pcb_info.txt`). Model preparation in Q3D Extractor usually requires some effort (in particular, all electrical terminals must be defined manually); here, this is taken care of automatically using a script (Vbscript), based on `pcb_info.txt`. The circuit model of the layout (parasitic elements) is then inserted in the electrical circuit of the converter and can be simulated using LTSpice

#### A. File formats for PCB layout export

Once a PCB layout has been designed in Altium Designer, its geometry has to be exported as a 3D-model. Several file formats have been introduced over the years for the description of a PCB layout, from the venerable "Gerber" (RS274-D, now RS274-X), introduced in 1980 up to the more recent ODB++

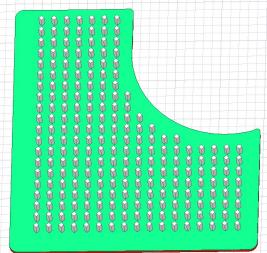
TABLE I: Comparison of the data transfer quality from Altium Designer to the Ansys suite.

	Gerber	ODB++	IPC-2581	EDB
3D-model	--	-	-	++
Layout details (materials,Nets)	-	-	+	++
Components details	-	-	+	++
Altium→HFSS layout	Yes	Yes	Yes	Yes
Altium→Siwave	Yes	Yes	Yes	Yes

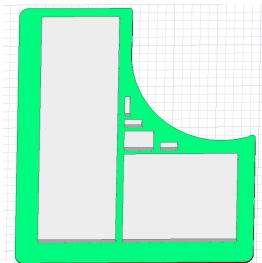
or IPC-2581. Each of these file formats offers a different set of features, with variations from one revision to another (3 revisions of the IPC 2581 format have been released so far), or from one implementation to another (some file formats include the possibility for proprietary extensions). Finally, the same variability can be observed regarding the "import" feature which will read the geometry description to turn it into an FE model.

As a consequence, it is difficult to predict the final quality of a layout description once it has been exported from eCAD and loaded into FEM without actually testing. Table I summarizes the results of the tests we performed using Altium Designer and the Ansys Electromagnetics suite. It shows that some file formats do not preserve the 3D geometry of a multilayer PCB (they only describe a set of 2D layouts, which must be ordered manually). Others do not describe the location of the components on the PCB. Finally, two pieces of software from the Ansys Suite can import a PCB geometry description: HFSS Layout and Siwave (Q3D Extractor does not provide an import feature for PCB geometry). Each of them can then generate a 3D model compatible with Q3D.

None of the standard file formats (Gerber, ODB++, IPC-2581) is found to allow a seamless data transfer. A fourth option, which relies on a proprietary format ("EDB", from Ansys) is found to offer better performance (most features are conserved, no error is observed). It requires the installation of an Altium plug-in provided by Ansys ("Ansys EDB Exporter", \*.def file extension). Export using the EDB file format, along with a file import through HFSS Layout is therefore used in this work.

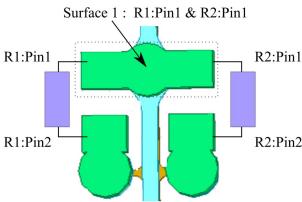


(a) A part of the 3D-model with 232 Vias

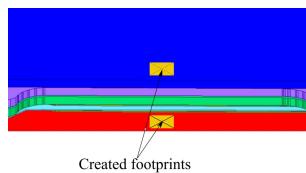


(b) The same object with the vias simplified to 6 boxes

Fig. 3: Advanced simplification: transforming the groups of vias into conductive boxes



(a) Pins simplification (surface 1 is  $\approx 2$  mm long)



(b) Created surfaces for the sink/source assignments

Fig. 4: Simplifying the number of terminals and creating footprints for detected faces with big area

## B. Model pre-processing, import and simplification

To complement the geometry description, some additional data are required. In particular, some pre-processing (a custom C++ code run from Altium) is required to locate the connection points for which the circuit parasitics must be computed. Indeed, only the pads which correspond to components should be used (some geometric features of layouts are detected as pads, but are not connected to an actual component). The references of the pads to be used are exported in a text file (`pcb_info.txt` in Fig. 1). This file, together with the geometry description (EDB file) constitute all the data used in the Ansys Suite to generate the 3D model (② in Fig. 2)

The `*.def` file is then imported into HFSS Layout, using a custom script. The first step ① is to locate the pads (x,y,z position), as this information is available in HFSS Layout, but it is not transferred to Q3D. Then the script uses a feature of HFSS to simplify the 3D geometry, by replacing curves with faceted polygons ②. This is particularly useful for copper vias, as these represent a large number of small features (thousands or tens of thousands of micro-vias can commonly be found in a multi-layer PCBs) which can be computationally intensive. Cylinders can be automatically replaced with prisms with a 4-, 12- or 24-sided base. The script produces a 3D-model which can then be directly processed using Q3D ③.

An additional simplification is presented in Fig. 3: vias are grouped together in large "boxes", removing most of the fine features (allowing for a coarser and lighter mesh of the model).

At the moment, this grouping requires manual interaction with the user.

## C. Ansys Q3D automation

In Q3D, the script defines the electrical paths for which it is necessary to calculate the circuit parasitics (resistance, inductance, capacitance, and all the mutual effects). In Q3D terms, this consists in defining "sources" and "sinks". Automating this avoids a tedious and error-prone manual entry of these sinks and sources. All objects composing the same conductor ("nets") are unified in a single part to reduce the model complexity and reduce simulation time. The automatic process also addresses two issues presented in Fig. 4: When a small surface hosts two or more connections, it is simplified to a single terminal (as it is the case for Surface 1 in Fig. 4a); on the contrary, small pads are defined on large conductor planes to accurately define connection points (by default Q3D affects entire surfaces to a single connection point), as presented in Fig. 4b.

Finally, the script runs the computation of the 3D model ④ and the generation of the corresponding Spice netlist ⑤ (`*.cir`).

The simplifications (especially that described in Fig. 4a) result in changing the number of terminals of the layout's Spice model (Pin 1 of R2 and pin 1 of R1 end up being connected to the same terminal of the Spice model, instead of being connected to two separate terminals). A description file (`spice_info.txt`) is generated to describe the interface between the schematic defined in Altium Designer and the Spice netlist.

## D. Ltspace schematic generation

Using the Spice model of the layout (③ in Fig. 2) and the schematic of the circuit as it has been designed by the user in Altium Designer ④, one last C++ Altium program is used to generate a LTSpice schematic (`*.asc` format). It reads the Altium schematic (fig. 5a) to get the connections between components, as well as the locations of the symbols on the schematic, so as to produce a readable output, in which all the connecting wires are replaced by their Q3D-calculated model (Fig. 5b). This model is too complex to be reproduced here. As an example, the data corresponding to the inductive part is presented in Tab. II.

This LTSpice schematic can then be opened in LTSpice. As a simulation is run, the software will generate the Spice netlist (exactly as it would do for a schematic that would have been entered manually). Note that the parasitic elements of the components (in particular the ESL/ESR of the capacitors) are integrated in their respective models. Here, we use the ESL and ESR values which are quoted in the datasheets of the capacitors to generate the capacitors models which are inserted automatically in the LTSpice netlist.

## E. Example

To test the workflow presented above, a simple test case is used: a half-bridge structure using IGBTs and diodes, rated

	C1-2	C2-2	D1-2	C1	D1-1	D2-2	R1-1	E1	C2	C1-1	C2-1	D2-1	R2-1	E2	R1-2	G2
C1-2	1.779	1.487	1.234	1.270	0.415	0.622	0.433	0.445	0.575	0.336	0.308	0.288	0.237	0.247	-0.012	0.009
C2-2	1.487	2.544	0.990	0.985	0.410	0.577	0.389	0.401	0.512	0.308	0.444	0.206	0.138	0.147	-0.011	0.009
D1-2	1.234	0.990	2.720	2.633	0.873	0.470	0.777	0.794	0.575	0.318	0.236	0.494	0.578	0.589	-0.015	0.010
C1	1.270	0.985	2.633	3.497	0.605	0.704	1.180	1.195	0.871	0.289	0.191	0.601	0.739	0.750	-0.015	0.011
D1-1	0.415	0.410	0.873	0.605	1.848	1.144	1.420	1.446	1.194	0.321	0.306	0.342	0.373	0.387	-0.023	0.014
D2-2	0.622	0.577	0.470	0.704	1.144	3.494	1.382	1.407	2.807	0.120	0.159	0.843	0.286	0.301	-0.023	0.016
R1-1	0.433	0.389	0.777	1.180	1.420	1.382	1.937	1.890	1.498	0.281	0.252	0.441	0.529	0.543	-0.016	0.014
E1	0.445	0.401	0.794	1.195	1.446	1.407	1.890	1.920	1.523	0.291	0.261	0.452	0.540	0.554	-0.026	0.014
C2	0.575	0.512	0.575	0.871	1.194	2.807	1.498	1.523	3.147	0.189	0.178	0.791	0.980	0.994	-0.023	0.015
C1-1	0.336	0.308	0.318	0.289	0.321	0.120	0.281	0.291	0.189	1.788	1.503	1.223	1.276	1.288	-0.009	0.011
C2-1	0.308	0.444	0.236	0.191	0.306	0.159	0.252	0.261	0.178	1.503	2.600	0.953	0.957	0.969	-0.009	0.011
D2-1	0.288	0.206	0.494	0.601	0.342	0.843	0.441	0.452	0.791	1.223	0.953	2.887	2.761	2.774	-0.010	0.013
R2-1	0.237	0.138	0.578	0.739	0.373	0.286	0.529	0.540	0.980	1.276	0.957	2.761	3.681	3.619	-0.010	0.005
E2	0.247	0.147	0.589	0.750	0.387	0.301	0.543	0.554	0.994	1.288	0.969	2.774	3.619	3.635	-0.011	0.014
R1-2	-0.012	-0.011	-0.015	-0.015	-0.023	-0.023	-0.016	-0.026	-0.023	-0.009	-0.009	-0.010	-0.010	-0.011	0.047	-0.001
G2	0.009	0.009	0.010	0.011	0.014	0.016	0.014	0.015	0.015	0.011	0.011	0.013	0.005	0.014	-0.001	0.047

TABLE II: Inductance matrix ( $L_{AC}$  [nH] for frequency = 500MHz) of the tested half bridge cell computed using Ansys Q3D Extractor with sources and sinks placed automatically. The diagonal element represents the inductance from the source to the sink on every net (5 nets= 5 sinks). The off-diagonal element represents the mutual inductance between the two nets.

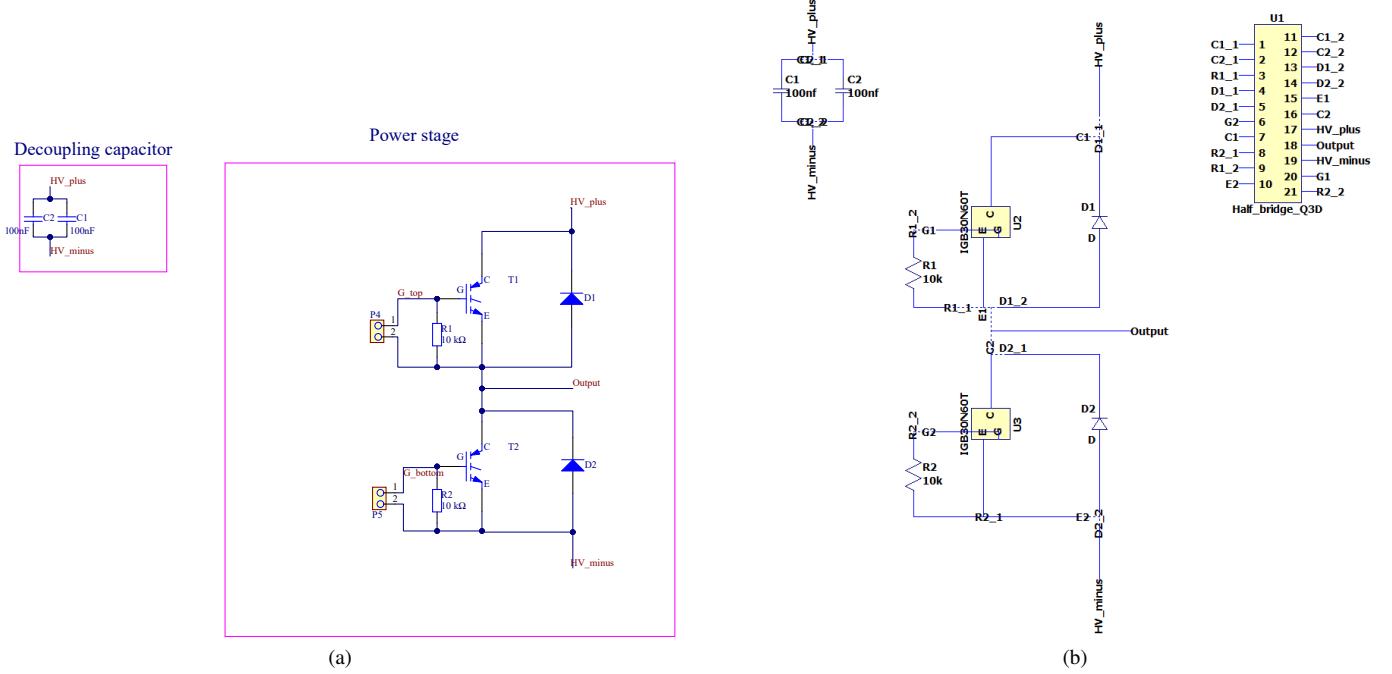


Fig. 5: (a) Altium schematic of the half-bridge structure used as an example and (b) LTSpice schematic generated automatically, using the Altium schematic to define the position of the components. U1 (top right) contains the layout's spice model (parasitic elements) calculated using Q3D Extractor.

at 600 V and 30 A). Despite its simplicity, the PCB contains advanced features: 6 layers, vias and microvias, 4 embedded chips).

The layout model is calculated following the steps described in sections II-A to II-C, without manual interaction. Pre-processing of the data and preparation of the 3D model only take a few tens of seconds, and most of the computation time is taken by the computations in Q3D Extractor.

As described in section II-B, some geometrical simplifications can be performed before computation. In particular, circular surfaces can be faceted, and groups of vias can be

replaced with a "box" (Fig. 3). Fig. 6 presents the results of such simplifications on the processing time, number of elements in the mesh, memory use, and maximum error. This error is calculated by comparing the R,L and C matrices generated by Q3D Extractor, on an element per element basis, with those of the case without simplification (cylinders remain cylinders); the value in Fig. 6 (bottom right) corresponds to the largest deviation.

Fig. 6 shows that systematic simplification (faceting of the circular shapes) offers a significant improvement in processing time and memory use (more than 50 % reduction for the 4-

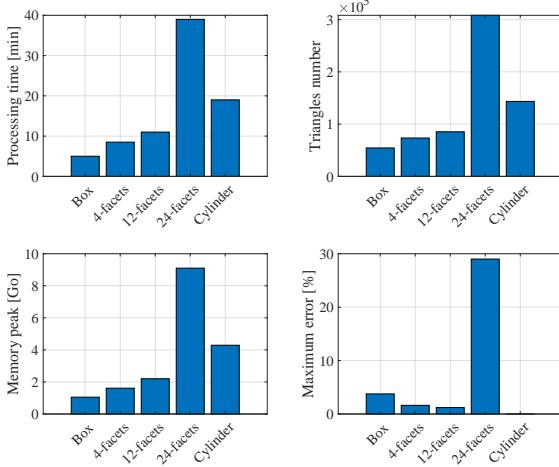


Fig. 6: Effect of the geometry simplifications on Q3D Extractor calculations (processing time, number of elements in the mesh, memory used, and maximum error compared to the case without simplification – cylinder –), for the half-bridge example presented in this article (on a Core i7 workstation with 16 Go RAM).

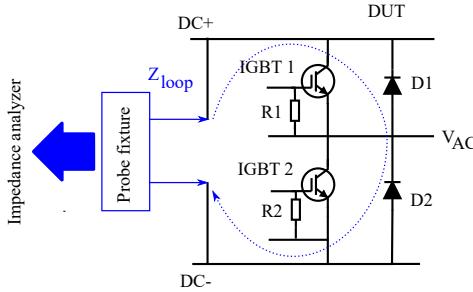


Fig. 7: Circuit diagram for the impedance measurement of the power loop.

facet case over the cylinder case), without increasing error noticeably ( $\approx 2\%$ ). Increasing the number of facets to 12 only marginally reduces the error. Further increase to 24 results in a dramatic degradation of all parameters, including error, probably because of some numerical instabilities. Box simplification offers shorter processing time and memory use as compared to the 4-facet case, at the cost of a slightly larger, but still acceptable, error ( $\approx 4\%$ ).

### III. EXPERIMENTAL VALIDATION

To validate the modelling procedure, two experiments are conducted, using the same half-bridge PCB as analyzed in section II-E: an impedance measurement, to validate the Q3D model, and a double-pulse test, to compare experimental waveforms with those predicted using the complete LTSpice model.

#### A. Circuit impedance

The power loop impedance of the PCB is measured using a Keysight E4990 impedance analyzer and a Keysight 42941A probe.



Fig. 8: Close-up on the PCB under test (Dimensions: 35mm\*45mm) and the tip of the Keysight 42941A probe placed across the decoupling capacitor footprint to measure the power loop impedance (decoupling capacitors not mounted).

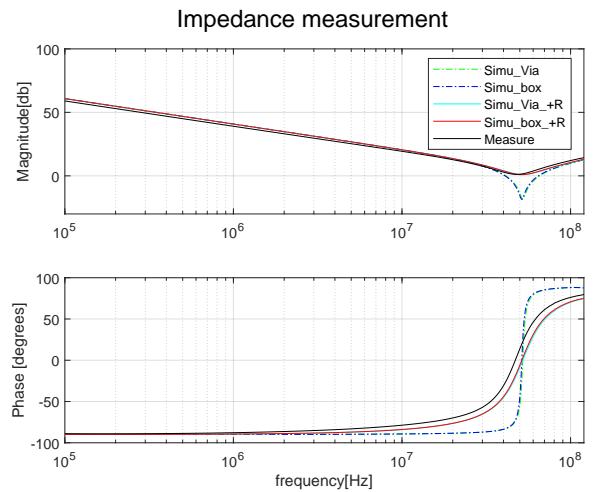


Fig. 9: Experimental and simulated bode plot of the power loop impedance

Impedance probe kit suited to in-circuit measurements. The measurement is performed with an amplitude of 10 mV and a frequency sweep ranging from 100 kHz to 120 MHz. Compensation are performed according to the user manual (phase, open and short compensations for the probe adapter, open and short for the tip of the probe). The embedded IGBTs are left un-controlled, with the pull-down gate resistors forcing gate-to-emitter voltage to zero. No decoupling capacitors are mounted on the PCB, and the impedance is measured across their (empty) footprint (Fig. 8).

In parallel, the experiment is reproduced in simulation, using the Spice schematic generated by the workflow described above, removing the decoupling capacitors, and replacing them with a 10 mV AC source. An AC simulation is then run, over the same frequency range as used experimentally. 2 configurations are simulated: no via simplification and "box" simplification.

The results are presented in Fig. 9. It can be seen that simulation results ("Simu\_via" and "Simu\_box") fit the exper-

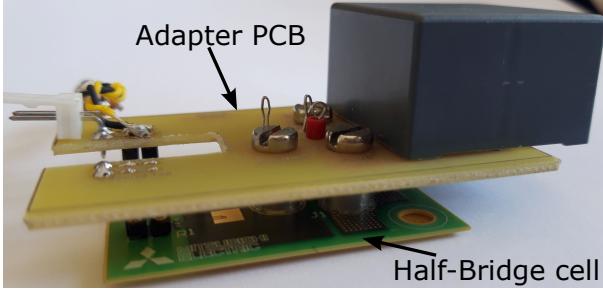


Fig. 10: The half-bridge cell (with two IGBT and two diodes embedded in the PCB, and two surface-mount decoupling capacitors). An adapter board is mounted on top for easier connection and to provide more decoupling capacitance.

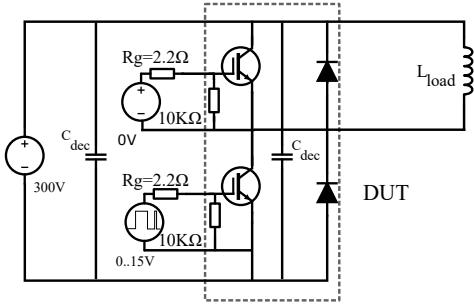


Fig. 11: Double pulse setup.

imental data satisfactorily, except at the resonance point (around 50 MHz). This can be corrected by adding a  $1\Omega$  resistor in series with the voltage source in the simulation file ("Simu\_via +R" and "Simu\_box +R"). This additional resistance is probably related to the probe/PCB contact: changing the probe position resulted in changes as high as 0.2 to  $0.3\Omega$  at the resonance point. On the contrary, changing the frequency value for which the layout parasitics are calculated in Q3D produced no noticeable change, even for values as high as 500 MHz. 4-point static characterization of the embedded IGBTs also showed that DC resistance is low (down to a few tens of  $m\Omega$ , dynamic resistance of the IGBTs included), so the high resistance at resonance cannot be associated to defects in the PCB's interconnects.

In conclusion, no noticeable difference can be observed between simplified and not simplified models. Simulated results show good agreement with the measurement (for example, simulated  $L_{loop} = 6.66\text{ nH}$  while experimental  $L_{loop} = 6.89\text{ nH}$ ). Further investigations are ongoing (using a new board allowing for 4-point impedance measurements) to validate the hypothesis that the resistance at the resonance point is only due to the probe-PCB contact.

#### B. Double pulse test

A standard double pulse test is built (Figs. 10 and 11) and reproduced virtually using LTSpice. Measurements are performed using passive probes (TPP0500B voltage probe & TCP0030A current probe, Tektronix) and a scope with

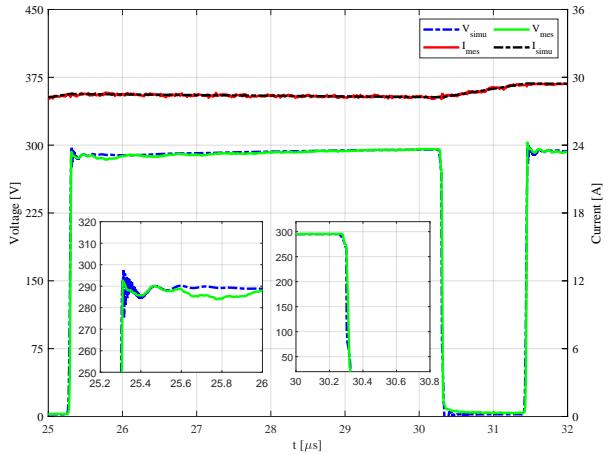


Fig. 12: Example of double pulse waveform

a bandwidth of 350 MHz (DPO4034B, Tektronix). Results are presented in Fig. 12. A good match is found between experimental and simulated waveforms.

#### IV. DISCUSSION

The modeling workflow presented here allows to automatically generate the SPICE model of a converter design in minutes, whereas manual model preparation would typically take days, with a strong chance of introducing errors in the connections. Because it is based on established software, advanced features of PCBs are taken into account (multiple layers, microvias, embedded chips). Only a few additional scripts are required to control model preparation and exploitation. The same approach is currently under development to generate the thermal model of a PCB.

The workflow has been tested on several designs (not presented here), and has been found to consistently produce satisfying models. For the largest models, which can contain more than 100 000 vias, the "box" simplification is mandatory. An algorithm is currently under development to remove the need for manual selections of the vias to be grouped. At the moment, no additional simplification has been considered (such as removing some of the tracks from the Q3D Extractor model), as our main goal is to allow for systematic and automatic model generation. However, as we investigate more and more complex designs, this approach may result in larger and larger models, and eventually become unpractical. It is not clear at the moment where this limit may be.

Another topic under development is the definition of "test benches" for the electrical and thermal simulation: while the PCB design in the eCAD software contains all the data needed to describe the board (part number, layout, materials, etc.), it does not provide any information regarding the "boundary conditions": input/output signals, loads, presence of a grounded heatsink, heat exchange coefficients on the surfaces, temperature... At the moment these must be entered manually, which can be burdensome in case of iterative design, and can cause

errors (mounting the PCB on a heatsink, for example, would change the parasitic capacitances and inductances, so it should be taken into account in the Q3D Extractor simulations). The approach being pursued in our work is to describe a "test bench" (a standard practice in VHDL designs). It consists in a text file which describes the relationship between the PCB and its environment.

## V. CONCLUSION

The workflow presented in section II allows to automatically go from the printed circuit board design of a power converter to a circuit model which includes circuit parasitics (a complete "virtual prototype"). Commercial software (Altium, Ansys Q3D, LTSpice) is used, with custom code at the interfaces. Some systematic simplifications of the geometry are proposed to accelerate computations, and the error they introduce is not found to be significant.

Experimental validation on a half-bridge PCB shows a good agreement between simulations results and measurements, regarding both impedance measurements and double pulse waveforms. Further investigations are ongoing, considering more complex designs and faster semiconductor devices (such as SiC MOSFETs, for which circuit parasitics cause more ringing).

## REFERENCES

- [1] T. Lagier, P. Dworakowski, L. Chédot, F. Wallart, B. Lefebvre, J. Maneiro, J. Páez, p. ladoux, and C. Buttay, "How Good are the Design Tools in Power Electronics?" in *EPE'20 ECCE Europe*, Lyon (Virtual conference), France, Sep. 2020. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-02968867>
- [2] P. L. Evans, A. Castellazzi, and C. M. Johnson, "Automated fast extraction of compact thermal models for power electronic modules," *IEEE transactions on power electronics*, vol. 28, no. 10, pp. 4791–4802, 10 2013.
- [3] J. Biela, J. Kolar, A. Stupar, U. Drozenik, and A. Muesing, "Towards Virtual Prototyping and Comprehensive Multi-Objective Optimisation in Power Electronics," in *Power Conversion and Intelligent Motion Conference (PCIM 2010)*, Nuremberg, Germany, May 2010, p. 23.
- [4] S. Yan, Z. Zhou, and V. Dinavahi, "Large-scale nonlinear device-level power electronic circuit simulation on massively parallel graphics processing architectures," *IEEE transactions on power electronics*, vol. 33, no. 6, jun 2018.
- [5] L. Boettcher, S. Karaszkiewicz, T. Löher, D. Manessis, and A. Ostmann, "3D modular power electronic systems, based on embedded components," *Advancing Microelectronics*, vol. 46, no. 1, pp. 6–10, 1 2019.
- [6] R. Caillaud, C. Buttay, R. Mrad, J. Le Lesle, F. Morel, N. Degrenne, and S. Mollov, "Thermal Considerations of a Power Converter with Components Embedded in Printed Circuit Boards," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2019. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-02291493>
- [7] A. Ostmann, L. Boettcher, D. Manessis, S. Karaszkiewicz, and K.-D. Lang, "Power modules with embedded components," in *Microelectronics Packaging Conference (EMPC) , 2013 European*, Sep. 2013, pp. 1–4.
- [8] C. Buttay, C. Martin, F. Morel, R. Caillaud, J. Le Leslé, R. Mrad, N. Degrenne, and S. Mollov, "Application of the pcb-embedding technology in power electronics – state of the art and proposed development," in *Proceedings of the International Symposium on 3D Power Electronics Integration and Manufacturing (3DPEIM)*, College Park, USA, jun 2018. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-01844981>
- [9] S. Hoffmann, E. Hoene, B. Schroeder, B. Stube, A. Alraai, O. Moritz, and O. Mueller, "Pcb layout tool integrated loss and inductance estimation," in *CIPS 2020; 11th International Conference on Integrated Power Electronics Systems*, 2020, pp. 1–5.