Journal of Evolving Space Activities
Vol. 1, Article No. 31, 2023
DOI: 10.57350/jesa.31

Snippet Based Electronics Design for Spacecraft Avionic Controllers

By Niklas AKSTEINER, [1)] René SCHULZ, [1)] and Janis Sebastian HÄSEKER[1)]

[1)] *Institute of Space Systems, German Aerospace Center (DLR), Bremen, Germany*

(Received July 27, 2022)

In this paper, we introduce an alternative approach for design, development and testing of electronics for space applications based on the reuse of tested and qualified design elements. Electronics for space applications have long development cycles due to mission specific design, part level qualification, functional verification and unit level testing. We try to reduce the recurrent work by reusing and testing design elements on circuit level. Associated with the recurrent work are overlooked design flaws and long test campaigns leading to risks for budget and schedule. To mitigate these shortcomings, we propose an alternative approach for development, implementation and testing based on the reuse of circuit-level layout elements, which we call Snippets. Besides schematic and PCB layout, simulation models, test results and documentation can be added on the same level. Limiting Snippets to a single function allows simple functional verification and maximizes reusability. The advantages of early testing include the elimination of parts unsuitable for the mission, validation of the expected functionality and verification of the PCB layout.

Key Words: Development Methodology, Design Automation, PCB Layout, Modularity

Nomenclature

COTS	:	Commercial Of The Shelf
CC/CV	:	Constant Current / Constant Voltage
DC	:	Direct Current
EDA	:	Electronic Design Automation
ECAD	:	Electronic Computer Aided Design
EGSE	:	Electrical Ground Support Equipment
EMI	:	Electro Magnetic Interference
EPS	:	Electric Power Subsystem
HDL	:	Hardware Description Language
IC	:	Integrated Circuit
I/O	:	Input and Output
LCL	:	Latching Current Limiter
LLRM	:	Latch Lock and Release Mechanism
MoCHI	:	Modular Circuit Hardware Integrator
PCB	:	Printed Circuit Board
PCDU	:	Power Conditioning and Distribution Unit
PEBB	:	Power Electronic Building Block
S/W	:	Software
TID	:	Total Ionizing Dose

1. Introduction

Modular design in electronics is not a new approach, but new methods to shorten the development cycle and to optimize the solutions are constantly evaluated.

In the field of power electronic the concept of Power Electronics Building Blocks (PEBB) was introduced for large scale naval systems in the 90s. It focusses on generalized switching components with integrated control and telemetry. The design driver being ease of replacement and reduction of recurrent design work. [1)] Recently this concept has been applied to PCB level design of optimized power converters over wide ranges of input and output conditions. A key feature is the design automation through serial and parallel connection of identical converter blocks, leading to reduced qualification effort and well understood circuit characteristics. [2)]

In commercial electronics the *Arduino*® and *Raspberry Pi*® ecosystems are examples of pre-designed circuits on Printed Circuit Board (PCB) module level enabling the design of complex systems from functional building blocks. Often used as a breadboard solution for prototyping, this approach allows early testing with relevant hardware and software. From there on, tools like Sparkfun® *À La Carte*® allow the maturation into basic application specific PCBs. [3)]

We seek to advance this concept of design reuse and modular design by applying it to a larger set of different circuits directly on PCB design level. Contrary to design reuse as a task for the board designer, we try to design single-function building blocks that can be matched as implementations to the top-down functional description. Following from this approach we explore the implications on testing, documentation and development methodology for aerospace systems.

2. Motivation

Electronic design can be a tedious and often repetitive task as many systems are made up of repeated, identical functions. Basic implementations are often reused across multiple projects, but rather on schematic level than on layout level. Firstly, this practice does not only introduce the risk of human error, but also takes up a lot of time, limiting the creative output of the designers. Reusing a finished, tested design eliminates these risks
and frees up designers for improvement instead of repetition.

Secondly the space environment poses additional requirements on electronics, like vacuum and high

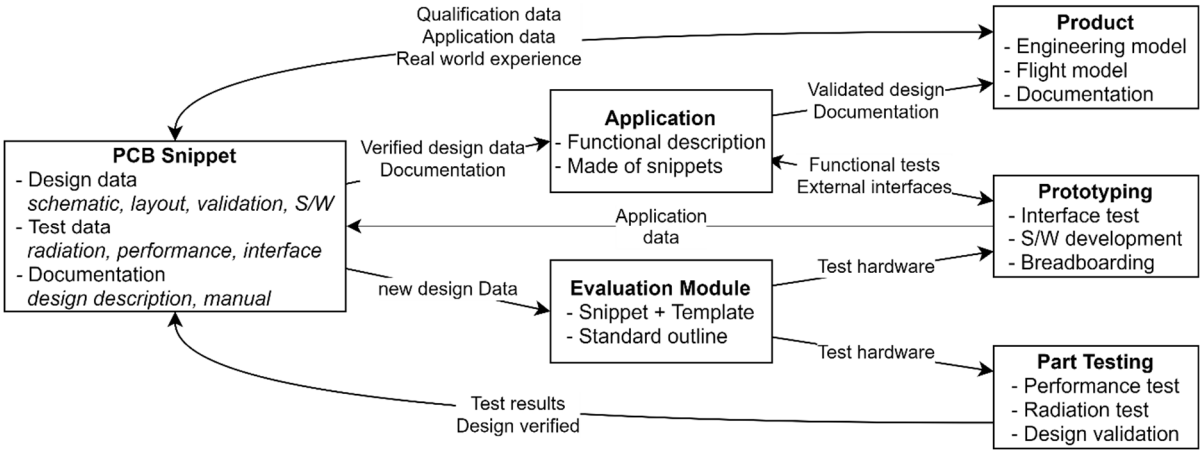

Fig. 1. Overview of snippet data and design flow for testing and final application.

temperature, but most critically radiation. To evaluate the effects of radiation on specific components, they are tested independently under different conditions. For these tests, a sample PCB has to be manufactured, which needs to be compatible with the test environment and similar to the final application to ensure the validity of the results. Although official evaluation boards are available in most cases, these can be problematic due to external components making failure analysis more difficult. A dedicated design is also often more cost effective, as multiple samples for each of the different test campaigns, like total ionizing dose, proton irradiation and heavy ion irradiation, are necessary.

A third aspect is the prototyping of electrical interfaces early in the project to support the development. For in-house developments the PCB modules enable a breadboard approach to ensure functionality and to identify emergent problems early on. They also allow the timely creation of small test boxes that can be given to external partners to check compatibility to the final hardware.

Design reuse on circuit-level layout elements promises to solve these problems, as it allows to quickly generate standardized evaluation boards for qualification and testing as well as providing reliable layout snippets for new designs. The different applications in different phases of the development are shown in Fig. 1.

3. Concept

We try to reduce the recurrent work by reusing and testing design elements on circuit level. Design reuse is a common practice in software and the field of functional programming already provides a general set of rules to design for reusability and testability. A central concept is the pure function, which only acts upon its input arguments and returns a result, without causing side effects in other parts of the program. A pure function can be easily replaced with its expected results without changing the function of the program.[4)]

In electronics the same concept can be applied to the design of a circuit. The basic setup is a strictly hierarchical schematic. Every electrical function (e.g. voltage conversion) is defined on its own schematic sheet and all external connections are explicitly made (no global labels). This has three consequences:

- Functions can be easily reused
- Functions can be tested independently
- The top-level schematic becomes a functional block diagram

A limiting factor for the reuse is the practical implementation of the circuit. On one hand for a certain schematic there are infinite component combinations as well as layout variations. On the other hand, every implementation provides a certain elasticity towards input and output variations. A DC-DC converter for example can be adapted to different output voltages by changing the inductor and resistor values, while keeping the same footprints. However, this elasticity always implies a deviation from the designated operating point leading to oversized solutions. A trade-off between additional design work and the non-optimal solution can be helpful.

It seems, that a layout is the strictest description of the implementation of a function, and thus it was chosen as the defining feature of a snippet. Consequently, multiple snippets with different component choices may relate to the same schematic. This means that for one function there may exist multiple snippets that can be used as implementation.

The following set of rules was found useful to generate reusable snippets and clarify their dependencies. These rules are kept very general, as the choice of Electronic Design Automation (EDA) software greatly influences how the concept is adopted.

Restrict functionality: A Snippet shall be limited to either the implementation of a single function or the combination of a single Integrated Circuit (IC) with its necessary additional components (e.g. a voltage regulator and its I/O capacitors).

Define interfaces: All connections of the Snippet need to be clearly named and described, just like the arguments of a function in programming. Naming

conventions enable automatic matching and convey more information. Interfaces should be routed to the outline of the snippet's PCB layout or to vias to allow easy connection later on.

Include documentation: All necessary information for the application of a Snippet should be part of the schematic. This includes, but is not limited to, the calculation of voltage dividers, set resistors or jumper configurations. Additional information can be handled in the database. Typical additional information would be simulation models, test results or previous applications.

Generalize board rules: Design rules as well as layer stack need to be consistent between Snippets to ensure compatibility for manufacturing. Alterations are possible, but care has to be taken. If rules are mixed, the most stringent set of rules is applied for the whole board.

On a higher level the restriction to one function per snippets means that the line between functional block diagram and hierarchical schematic gets blurred. A similar observation can be made for a board level Hardware Description Language (HDL).[5] The HDL allows a functional description of the schematic, similar to a block diagram of the general architecture. This high-level description is augmented through detailed implementation schematics called refinements.

4. Application Example

To show the practical implications of this design approach, the following chapter describes the usage of a single snippet from design and part level testing over prototyping to a finalized product. The required function of the snippet is voltage conversion with output current regulation for the deployment of the landing legs of the joint JAXA, CNES and DLR re-usable launcher demonstrator CALLISTO. [6]

4.1. Snippet design

The design process starts out with the selection of the central part (e.g. DC-DC converter IC), the creation of a schematic and the design of a generic layout. Parts of interest can be selected even before a specific application is planned, as certain functions like DC-DC converters are widely used. The converter IC used for this snippet was selected due to the versatile current regulation loop but without a specific application in mind. The first snippet with this part is designed for maximum versatility. The placement of different jumper resistors allows the configuration for fixed output voltages, fixed output current or Constant Current/Constant Voltage (CC/CV) operation. The footprints are chosen to accommodate capacitors and inductors with high voltage and current ratings, resulting in the largest expected packages. The resulting design is adaptable to many applications and allows testing of all functions. Fig. 2. a) shows the schematic representation on the hierarchical sheet with its input and output signals. Fig. 2. b) shows the basic layout used for this snippet. Note the outline around the snippet. All parts, copper pours and lines inside of it are part of the snippet. The outline is a help during the layout phase but also allows a rough estimation of the arrangement later on without moving all components.

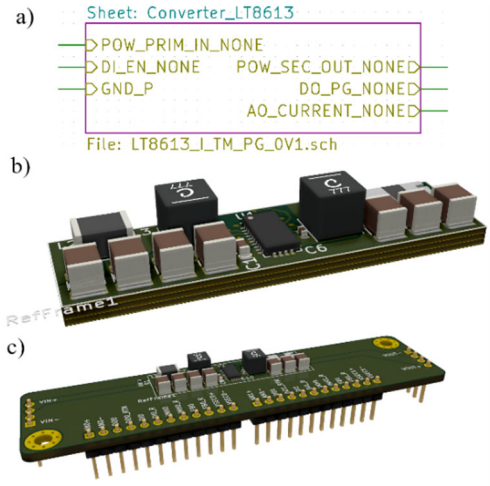

Fig. 2. Views of a snippet: a) schematic, b) layout and c) evaluation module.

4.2. Module design

To minimize the manual work for testing, we designed a standardized template for evaluation modules. The preparation of a module is limited to connecting the hierarchical sheet to the connectors of the module template. In some cases, adding pull-up/-down resistors simplifies the test setup. The layout is equally simple, as all snippet connections are either available on the outline of the snippet or on vias to allow connections on unused layers. Driven by the requirements of efficient packing during radiation tests, we decided on a 23 mm x 90 mm module. Typical interfaces of power controller circuits resulted in a main input and output connector and a set of command and telemetry lines. Fig. 2. c) shows the new snippet in this evaluation module formfactor with the main power pin headers on the short sides and the data and command pin headers on the long side. All electrical connections are specified to allow drop-in compatibility between different implementations of the same function. The mechanical outline can

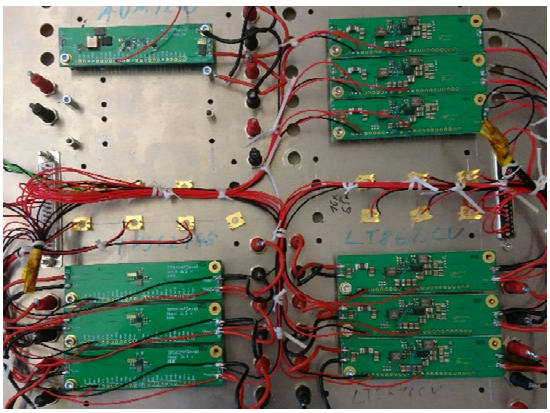

Fig. 3. Radiation test setup with three samples per configuration.

accommodate most snippets, but can be expanded on the connector-free top edge if necessary. The standardized mechanical interface and pin positions allow reuse of test fixtures, which in turn justifies the development of more complex fixtures to aide quick testing and characterization.

4.3. Evaluation module testing

With the evaluation module finished, we have a piece of hardware for testing in the real world. Closing the PCB design loop on this level enables quick iteration on single functions at minimum cost. It also means that every circuit gets verified before it is actually deployed in a larger project. Beyond the basic functionality, the module gives a first glance of the circuit's performance metrics like converter efficiency or current sensor accuracy.

Furthermore the module is used for radiation testing of the used components. Fig. 3. shows the test setup for Total Ionizing Dose (TID) testing with the modules mounted on a metal plate and harness for online characterization. This configuration was also used for proton irradiation testing. For TID offline characterization a quick-change test fixture with pogo pins to connect to empty pin header contacts is used. This shortens characterization significantly compared to a pin header and wire setup. As previously noted, these fixtures can be reused for all future evaluation modules.

Another valuable side effect of limiting snippets to single functions or ICs is the easy debugging of radiation effects due to the limited number of components and therefore reduces number of failure scenarios.

4.4. Module prototyping

As all previous work is carried out without a specific application field in mind, the designed snippets provide a head start as soon as real-world applications appear. In case of the CALLISTO landing leg deployment, a regulated current of 4 A into a 2 Ohm load is required for the Latch Lock and Release Mechanism (LLRM).

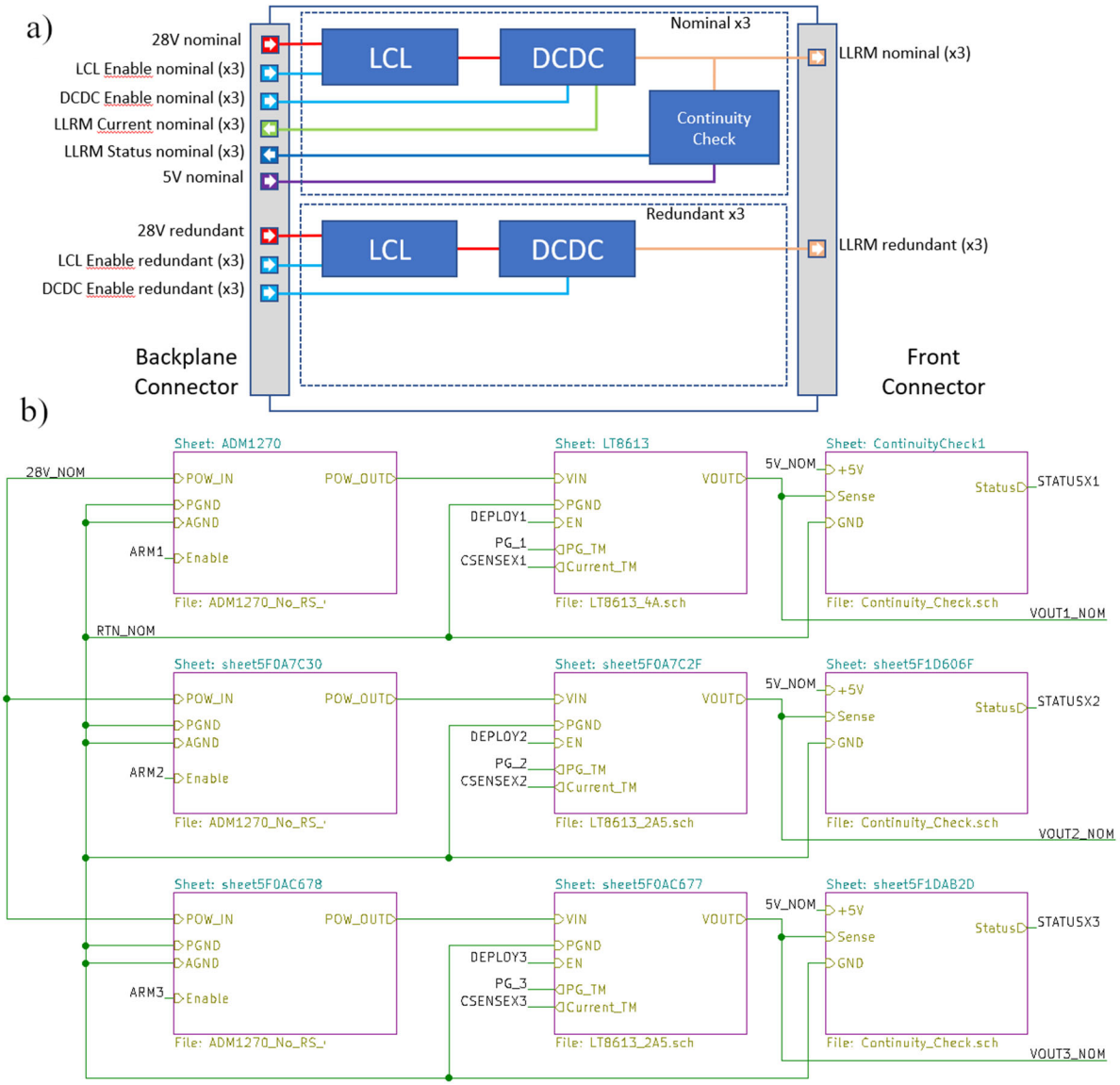

Fig. 4. a) Block diagram of LLRM driver board and b) schematic top sheet of nominal side of LLRM driver board.

On the supply side the input is a 28 V unregulated bus. This is well within the ratings of the converter IC we already have in our snippet library. The test module enables a quick preliminary test without any new hardware or waiting for PCB production. An evaluation module is configured for the required output current and voltage ratio while a prototype of the deployment actuator is already available. Within a few days a single channel of the deployment controller is tested with the actuator to validate the choice of solution. Preliminary tests show non-compliance with an inrush current requirement during turn on. This is mitigated through an additional current limiting module on the input side. The additional snippet is already in the database and the evaluation module available. The current limiter also acts as additional safeguard signal to prevent actuation through a single signal. The solution composed of two evaluation modules is built into a small enclosure and provided to the department in charge of the landing leg development and test campaign.

For more complex setups like power distribution units or complete PCDUs a modular breadboard is available to quickly assemble a set of modules into a functional breadboard solution.[7)]

4.5. Application specific board design

With the interfaces to the landing leg and the power bus tested the design of an integrated prototype starts. Each landing leg has three Latch Lock and Release Mechanisms with redundant inputs, thus six outputs per landing leg are needed. The mechanical outline for the board is already fixed but the connectors are still preliminary at this point. A generic set of connectors is selected and a board template generated to accept the snippets. A rough estimation from the size of the snippets suggests, that six channels can be placed on a single board. The layout of six identical deployment channels with inrush current limit is achieved by importing the two snippets into the project, cloning them and placing them on the board. The adaptation to two different current levels is achieved through component variation in the same layout. An additional snippet for output telemetry is generated specifically for this application at this point. Since the circuit is simple and another iteration of the board is necessary anyway, it was included without previously generating an evaluation module. Fig. 4 shows the block diagram (a) for the LLRM driver alongside the top-level schematic sheet (b), highlighting the similarity and clarity of the top-level circuit design.

Similar to 4.2. the main effort in designing the board is in routing the individual signals from the snippets to the connectors. Fig. 5. shows the first prototype without its connectors installed. After an initial test, it was placed in an enclosure and used for deployment tests of a complete landing leg. The inside of the enclosure is shown in Fig. 6. Note the secondary voltage supply, that is emulated by another converter

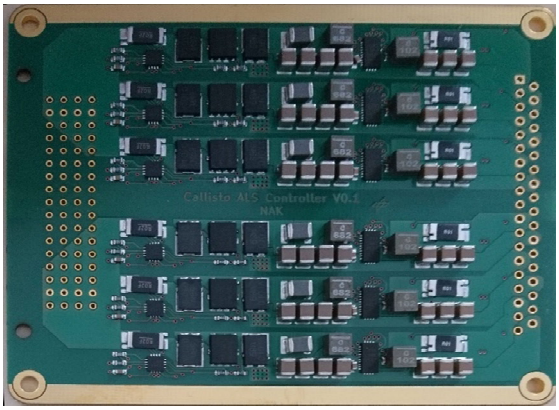

Fig. 5. First integrated prototype of LLRM driver module.

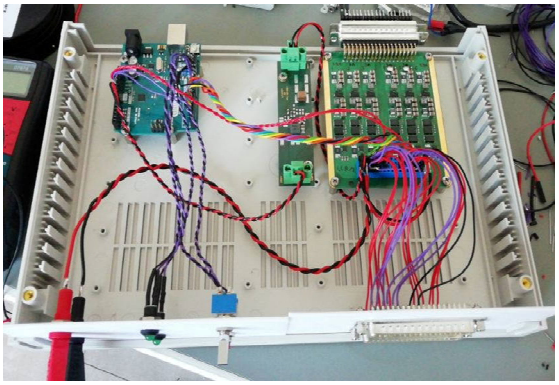

Fig. 6. LLRM driver box for mechanical landing leg tests.

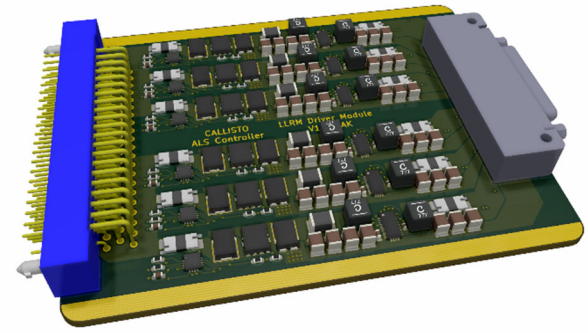

Fig. 7. Final design of LLRM driver module.

evaluation module in the middle. The microcontroller on the left is used to create the timing for the individual channels. This solution is identical to the final solution in terms of function, outline, telemetry and command interface. It proves that no emergent effects due to the integration of multiple snippets are present. Building on this success, only the connectors were changed for the final version as shown in Fig.7.

5. Future Activities

As design automation in electronic design is a quickly evolving topic and we discussed our approach with many colleagues, a vast set of ideas for further optimization potential came up. The following three points were selected due to their great significance in our everyday work.

5.1. Applications in data handling

Similar to power distribution, the data handling system needs varying application specific interfaces to connect to different external units or sensors. Many of these interfaces are identical across many solutions and new solutions could in theory be generated from a functional description and snippet library. Compared to the power distribution, thermal considerations will be less critical for these circuits, but they can be more susceptible to crosstalk, EMI or other emergent effects.

5.2. Generator scripts

Our current workflow stops after the generation of the final layout, but due to the reuse of identical snippets fails to capture some alterations made on the component values. An interesting next step would be the generation of specific schematics with updated component values and layouts from a generalized snippet. Generally, a data model to allow efficient design reuse combined with easy alterations and forward/backward annotation needs to be developed to further minimize manual work.

5.3. Design automation

In order to further increase the degree of automation and to obtain even faster and better board layouts in the future, a tool for automatic layout optimization, the Modular Circuit Hardware Integrator (MoCHI) is currently being developed. MoCHI ties into our Design Automation Toolkit,[8] which can find and optimize different implementation options for a power controller unit based on high-level system requirements and a database of Snippets. The implementation options are exported as directed graphs, where a node represents a Snippet and the arrows indicate the electrical connection between the Snippets.

MoCHI then gets the appropriate directed graph, loads the necessary Snippets from the database, and automatically creates schematic and layout files in KiCad. In order for MoCHI to evaluate and move Snippets, characteristic data about each Snippet is collected in the database. For example, the position of the component with highest thermal losses or the position of the inputs and outputs within the Snippet are stored. Once all Snippets have been placed onto the board, MoCHI can calculate an optimality metric for the board based on these characteristic points. By moving Snippets and recalculating the factor, MoCHI can optimize the board towards the best achievable metric. The movement of Snippets is based on rules that a developer intuitively applies during the layout process and that have been implemented here as algorithms. One rule is, for example, that the board should be designed thermally well, which means that Snippets with high losses must be placed as close as possible to the next low resistance thermal interface. Within a GUI, the user can then influence the optimization process by giving higher or lower weights to individual rules. While the schematic is strictly given by the directed graph, there are many different design possibilities for the board layout based on the weight distribution. Through the reuse of PCB snippets and an auto-router (e.g. free-routing) a complete PCB layout including ERC check can be created.

Currently, two rules are implemented in MoCHI, one is the thermal and the other is the electric rule, which wants to keep PCB traces between Snippets as short as possible. The graph and the wiring refer only to the power connections, the treatment of the signal lines is planned for the future. The next steps here are the optimization of the software and the first use in a larger project.

6. Conclusion

A general framework for the structured design reuse of PCB layouts is presented and demonstrated on a first example application. The approach ensures the design verification of layout elements and accelerates supporting testing activities through the generation of functional PCB modules. Furthermore, it improves the development speed of complex boards since incremental development from an existing base of snippets is encouraged.

Acknowledgements

Although no software specifics were given in the paper we wish to thank the creators and maintainers of KiCad for their incredible work on an open source EDA tool. [9]

Furthermore, we wish to thank Mitja Nemec for the creation of the restore layout plugin, which is at the backbone of this paper and enables the script-based snippet placement. [10]

References

1) Ericsen, T. and Tucker, A.: Power Electronics Building Blocks and Potential Power Modulator Applications, Conference Record of the Twenty-Third International Power Modulator Symposium, 1998, pp. 12-15., doi: https://doi.org/10.1109/MODSYM.1998.741179
2) Lamorelle, T., Andreta, A., Lembeye, Y., Crébier, J.-C., and Podvin, J.-C.: Design level power electronics building block: Industrial framework for DC-DC conversion, IEEE International Conference on Industrial Technology (ICIT), 2018, pp. 670-675, doi: https://doi.org/10.1109/ICIT.2018.8352258.
3) Sparkfun, Á la Carte PCB designer, https://alc.sparkfun.com/ (accessed December 5,2021)
4) Chiusano, P., Bjarnason, R.: *Functional Programming in Scala,* Manning Books, New York, 2015, pp. 3-10.
5) Lin, R., Ramesh, R., Jain, N., et al.: Weaving Schematics and Code: Interactive Visual Editing for Hardware Description Languages, The 34th Annual ACM Symposium on User Interface Software and Technology. Association for Computing Machinery, New York, NY, USA, 1039–1049. doi: https://doi.org/10.1145/3472749.3474804
6) Dumont, E., Ishimoto, S., Tatiossian, P., Klevanski, J., und Reimann, B., Ecker, T., Witte, L., Riehmer, J., et al.: CALLISTO: A Demonstrator for Reusable Launcher Key Technologies*, Trans. JSASS*, Aerospace Technology Japan, 19 (1), pp 106-115. doi: https://doi.org/10.2322/tastj.19.106
7) Häseker, J., Strowik, C., Aksteiner, N. and Dannemann, F.:

Design Process and Modular Breadboard for Rapid Development of Mission Specific Power Conditioning and Distribution Units, Poster SSC19-P1-16, Small Satellite Conference, Utah State University, Logan, UT, USA, 2019.

8) Häseker J. S., Aksteiner N.: (2020) Collaborative Design Automation Toolkit for Power Controllers in Aerospace Applications. In: Luo Y. (eds) Cooperative Design, Visualization, and Engineering. CDVE 2020. Lecture Notes in Computer Science, vol 12341. Springer, Cham. https://doi.org/10.1007/978-3-030-60816-3_4.
9) KiCad EDA, https://www.kicad.org/ (accessed December 5, 2021).
10) Nemec, M.: KiCad Action Plugins, https://github.com/MitjaNemec/Kicad_action_plugins (accessed October 20,2021).