DOI: 10.15514/ISPRAS-2025-37(2)-2

FOSS Tools Usage for Circuit Simulation in Analog IC Design

V.V. Kuznetsov, ORCID: 0000-0002-5738-555X <vadim.kuznetsov@bmstu.ru>
D.V. Andreev, ORCID: 0000-0001-9080-1028 <dmitrii_andreev@bmstu.ru>
G.K Grishin, ORCID: 0009-0002-1279-4699 <grishin_gr@mail.ru>

Elelctronic engineering department, Bauman Moscow State Technical University, Building 1, 5, 2nd Baumanskaya st., Moscow, 105005, Russia.

Abstract. Circuit simulation is an important part of the microelectronics design flow. This paper presents a design flow using Qucs-S circuit simulation tool with Ngspice or QucsatorRF backend. This solution may be targeted to both academic and industrial applications. Qucs-S combines a modern graphical user interface and a computation power of open-source circuit simulation kernels like Ngspice, QucsatorRF, and XYCE. The article provides an overview of Qucs-S software architecture and its application for integrated circuits design. The proposed workflow is illustrated by example of the semiconductor resistors parameter extraction and corner analysis using the mentioned open-source tools.

Keywords: computer-aided design (CAD); electronic design automation (EDA); microelectronics; circuit simulation; compact modelling.

For citation: Kuznetsov V.V., Andreev D.V., Grishin G.K. FOSS tools usage for circuit simulation in analog IC design. Trudy ISP RAN/Proc. ISP RAS, vol. 37, issue 2, 2025, pp. 19-32. DOI: 10.15514/ISPRAS-2025-37(2)-2.

Acknowledgements. The research was financially supported by the Ministry of Science and Higher Education of the Russian Federation as a part of the project FSFN-2024-0086.

Применения программного обеспечения с открытым исходным кодом для схемотехнического моделирования при разработке аналоговых интегральных микросхем

В.В. Кузнецов, ORCID: 0000-0002-5738-555X <vadim.kuznetsov@bmstu.ru>
Д.В. Андреев, ORCID: 0000-0001-9080-1028 <dmitrii_andreev@bmstu.ru>
Г.К. Гришин, ORCID: 0009-0002-1279-4699 <grishin_gr@mail.ru>

Кафедра проектирования и технологии производства электронной аппаратуры, Московский Государственный Технический Университет им. Н.Э. Баумана, Россия, 105005, Москва, 2-я Бауманская ул., д. 5, с. 1.

Аннотация. Моделирование схем является неотъемлемой частью процесса проектирования микроэлектронных устройств. В статье представлен процесс проектирования с использованием инструмента для моделирования Qucs-S с Ngspice или QucsatorRF в качестве движка моделирование. Данное ПО может быть использовано как в академических, так и в промышленных целях. Qucs-S сочетает в себе современный графический интерфейс и движки для моделирования, такие как Ngspice, QucsatorRF и XYCE. В статье представлен обзор программной архитектуры Qucs-S и ее применения для проектирования интегральных схем. Предлагаемый рабочий процесс проиллюстрирован на примере извлечения параметров полупроводниковых резисторов и углового моделирования с использованием упомянутых инструментов с открытым исходным кодом.

Ключевые слова: автоматизированное проектирование (CAD); автоматизация проектирования электроники (EDA); микроэлектроника; моделирование микросхемы; компактное моделирование.

Для цитирования: Кузнецов В.В., Андреев Д.В., Гришин Г.К. Применения программного обеспечения с открытым исходным кодом для схемотехнического моделирования при разработке аналоговых интегральных микросхем. Труды ИСП РАН, том 37, вып. 2, 2025 г., стр. 19–32 (на английском языке). DOI: 10.15514/ISPRAS–2025–37(2)–2.

Благодарности: Работа выполнена при финансовой поддержке Министерства науки и высшего образования РФ в рамках Госзадания FSFN-2024-0086.

1. Introduction

Circuit simulation is an important part of the electronic hardware design flow in the modern industry. The first circuit simulator called SPICE [1] was developed by Laurence Nagel from Berkeley University in 1973. The most of the modern circuit simulation software share the SPICE netlist syntax and its computation algorithms. The electronic components vendors are sharing the SPICE models for their integrated circuits (IC) and discrete devices. So, the SPICE became an industrial standard for circuit simulation and models interchange. There exists a number of open-source SPICE-compatible circuit simulation backends developed by international community. The most famous are Ngspice (licensed under BSD) or XYCE (licensed under GPL). The both tools operate in the command line (CLI) accepting the plain text netlist as input and providing the text or binary dataset as the output. Ngspice provides a postprocessing language called Nutmeg for manipulation with the simulation data. The both CLI tools mentioned above have a good simulation speed, performance, and device models compatibility and therefore recommended by Google for their free Skywater 130nm process design kit (PDK) for IC design. There is no need to develop another open-source simulation engine from scratch.

But the CLI circuit simulation tools are lacking the graphical user interface (GUI). Development of the unified GUI that allows fast simulation backend switching is an important task in free open-source (FOSS) circuit simulation tools development. Besides proprietary circuit simulation software

there exists a number of open source or freeware solutions with graphical user interface (GUI) summarized in the Table 1.

The table compares a number of existing circuit simulation tools. Only freeware proprietary and FOSS tools are compared. The MicroCAP circuit simulation tool is abandonware, because the Spectrum Software company has discontinued its development, but it remains free to download. The main disadvantage of both Ltspice and MicroCAP is that they are Windows-only. The KiCAD is not a circuit simulation software, but PCB design suite. But KiCAD contains extensions that allow to simulate schematic using Ngspice. Xschem is recommended by Google as the Ngspice frontend for their 130nm PDK. The main advantage of this tools is that it allows unlimited hierarchical cells design that is important for microelectronics. Other tools may have hierarchical levels limitation. No one of the freeware and open-source circuit simulation tools provide an advanced features for the radio frequency (RF) circuit simulation and compact modelling features. These features are implemented in the simulation backend level. There exists a need to develop a tool that brings these features to the GUI level.

Table 1. Circuit simulation tools comparison.

	Qucs-S	KiCAD	MicroCAP	LTSpice	XSchem
License	GPL	GPL	Abandonware	Freeware	GPL
Platform	Cross-platform	Cross-platform	Windows	Windows	Cross-platform
Simulation backends	Ngspice, Xyce, QucsatorRF	Ngspice	Proprietary SPICE	Proprietary SPICE	Ngspice
Hierarchical design	Subcircuits	Subcircuits	Subcircuits	Subcircuits	Unlimited cells
RF extensions	S-parameter; Transmission lines models; Smith charts	Not supported	Not supported	Not supported	Not supported
Postprocessor	Nutmeg, Qucs equations	No postprocessor	Proprietary	Proprietary	Nutmeg
Export to PCB	Not supported	Implemented	Not supported	Not supported	Not supported
Compact modelling	Verilog-A support using OpenVAF; Verilog-A synthesizer	Not supported	Not supported	Not supported	Not supported
Simulation backend switching	Quick switch from GUI	Not supported	Not supported	Not supported	Not supported

Qucs-S is based on the “Quite universal circuit simulator” Qucs [2], [3] project that was established by M. Margraf and S. Jahn in 2001. The Qucs was originally designed as radio frequency (RF) and microwave circuit simulator and had its own SPICE incompatible simulation kernel called Qucsator. This simulation engine had a number of unique features like scattering matrix and harmonic balance

analysis. Later the Qucsator was extended for simulation of general-purpose circuits but its time-domain analysis performance remained very poor compared to kernels based on SPICE. The Qucs-S [4] was forked from original Qucs project around 2017. Qucs-S is distributed under genaral public license (GPL) version two and cross-platfrom. It can operate on Linux, Windows, and FreeBSD platforms.

The purpose of Qucs-S fork was to bring SPICE compatibility, because SPICE netlist syntax remains an industrial standard. Qucs-S has switched from Qucsator to Nsgpice [5] opens-source simulation kernel as default simulation kernel to provide SPICE compatibility. Another open-source SPICE compatible simulation kernel XYCE [6] could be selected as optional.

The QucsatorRF simulation kernel ist kept for RF and microwave circuits analysis [7]. It is distributed with Qucs-S package and doesn't require a separate installation. The QucsatorRF is fork of the abandoned Qucsator simulation kernel. QucsatorRF is maintained by Qucs-S developers now. This simulation kernel is not SPICE-compatible and uses netlist syntax different from SPICE. The internal architecture of the QucsatorRF allows to define S-parameter matrix of the device in the C++ level model. This simulator provides for example microstrip lines models taking into account frequency dependency effects and dispersion. These models are not available for SPICE.

Besides a GUI for command line SPICE simulation kernels Qucs-S provides an advanced compact modeling features like Verilog-A models synthesizer [6] and OpenVAF compiler integration that allows using it for academic applications.

2. Tool architecture overview

The architecture of the Qucs-S is shown in the block diagram Fig. 1. The tool consists of the schematic capture and visualization system that operates on GUI level and SPICE circuit simulation engine that operate on the CLI level.

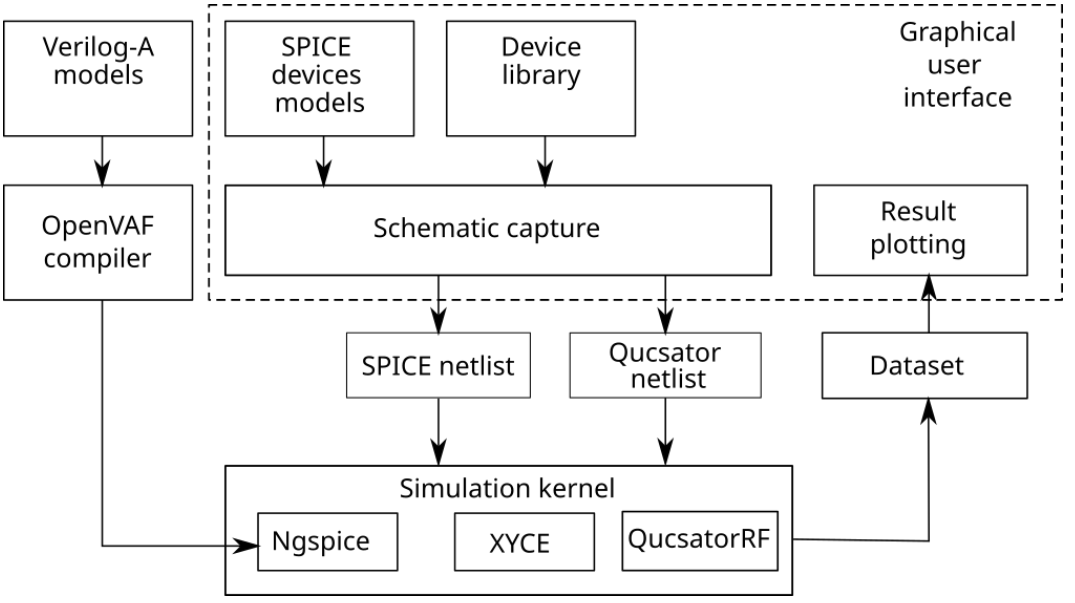

Fig. 1. Qucs-S software architecture.

A user enters circuit data using the GUI schematic capture, and then the tool automatically generates netlist that is passed to Ngspice or XYCE circuit simulation kernel. Qucs-S doesn't contain an internal SPICE engine, but provides a user interface to external SPICE-compatible simulators. The legacy Qucsator simulation kernel and its netlist syntax also could be selected for backward

compatibility. After the simulation is finished the waveform data is saved to dataset which is parsed by the GUI to plot the waveform. Qucs-S supports not only SPICE models but also Verilog-A models that are could be linked to SPICE simulation kernel using the open-source OpenVAF compiler [8].

The main window of the Qucs-S application is shown in the Fig. 2. Arrows are not the GUI design elements, but indicate the key elements of the interface. Plain text XML format is used for saving schematic and libraries. This allows to write an extensions like schematic synthesizers. The devices, equations, simulations and diagrams are represented as the graphical objects that are placed on the schematic field (pointed by arrows, see the Fig. 2). It is possible to build hierarchical schematics using subcircuits and library devices. Qucs-S allows users to construct SPICE device definitions from a name, a model specifier and a SPICE style modelcard. These can be attached to a schematic symbol and passed directly to a SPICE kernel. Fig. 3 illustrates the construction of a SPICE netlist entry for a bipolar junction transistor (BJT) device. Arrows indicate the connection between the schematic editor GUI elements and SPICE netlist directives.

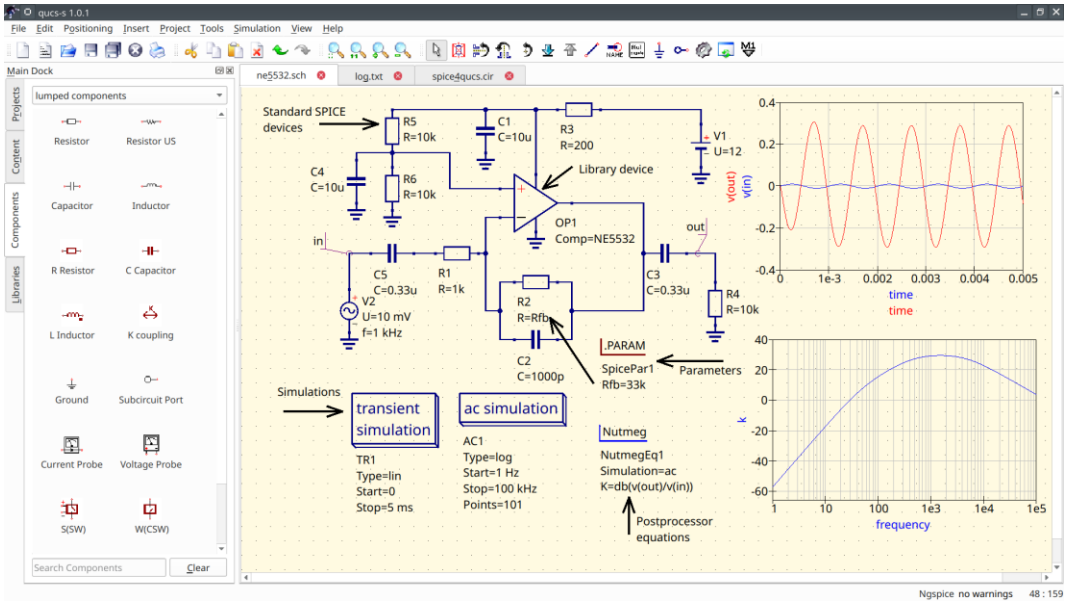

Fig. 2. Qucs-S application main window.

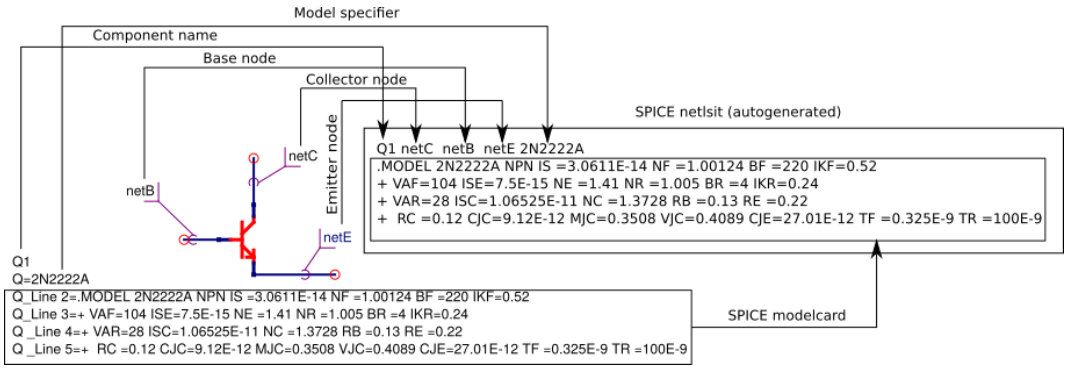

Fig. 3. Construction of a SPICE netlist entry for BJT.

Qucs-S is cross-platform application and is written in C++ using Qt6 toolkit. The block diagram representing the C++ classes hierarchy and its interaction is shown in the Fig. 4. The components

representing devices and simulations are represented as the C++ classes. The class properties are holding the SPICE parameters. The netlist generator is also represented as C++ class. Ngspice and XYCE both are SPICE-compatible but has some difference in netlist syntax. The common SPICE netlist routines are implemented in the AbstractSpiceKernel class. The simulation kernel specific is implemented in the inherited classes. The interaction between the simulation kernel and GUI is implemented using the standard Qt signal/slot mechanics. Qucs-S supports the mechanics of quick switching of the simulation kernel without of the restart of the application.

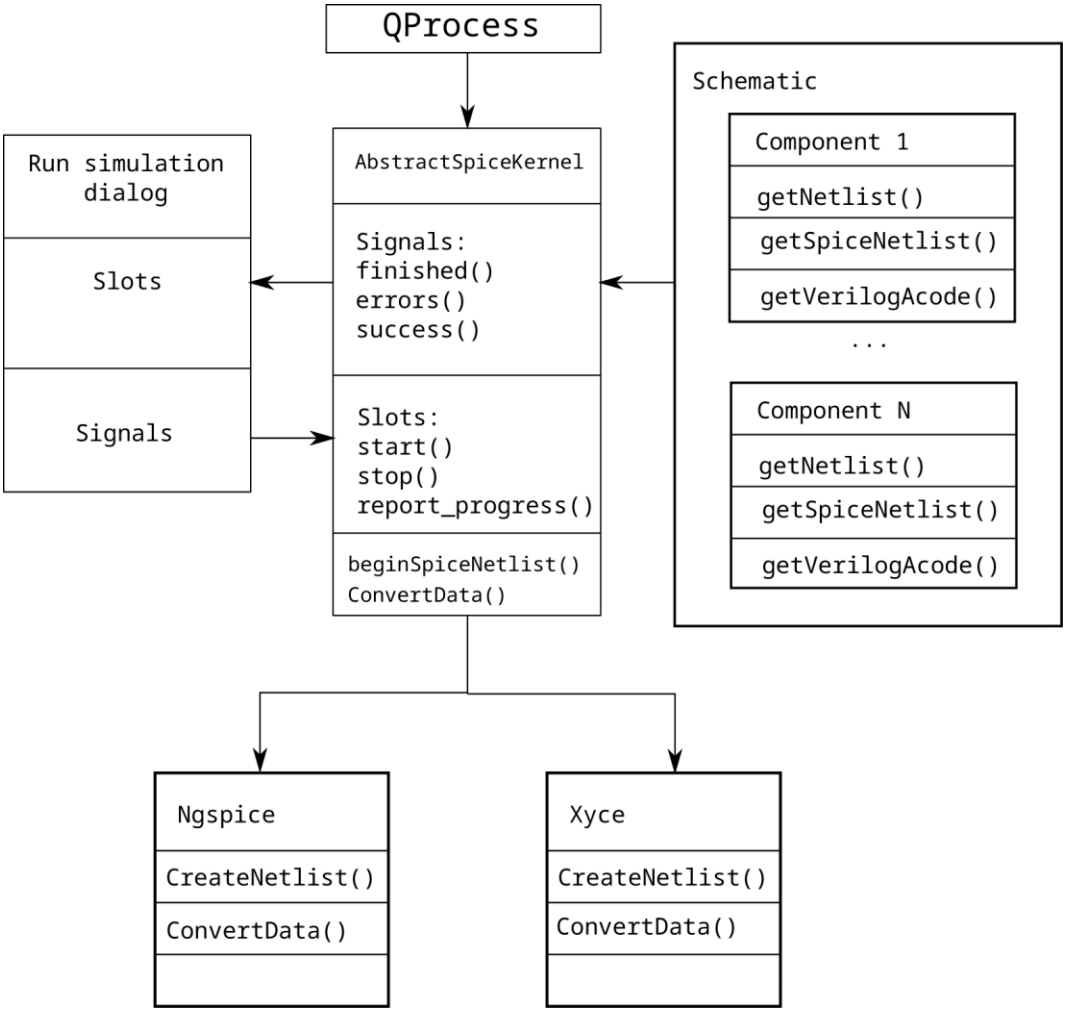

Fig. 4. Qucs-S C++ classes inheritance and interaction diagram.

Qucs-S provides a set of system library devices designed to provide a support of microelectronic process design kits (PDK) [8]. These library devices allow to assign SPICE letter, pin number, modelcard name, and model parameters. Qucs-S supports the simulation using both Google Skywater 130nm [9] and IHP open BiCMOS [10] PDKs. The Fig. 5 illustrates the integration of Google Skywater 130nm models in Qucs-S schematic document. The MOSFET models are taken from the library that is indicated by arrow. The Ngspice is used as the simulation kernel [11]. Besides the PDKs support Qucs-S has a number of RF simulation features like transmission line models, S-parameter and harmonic balance simulation, Smith charts.

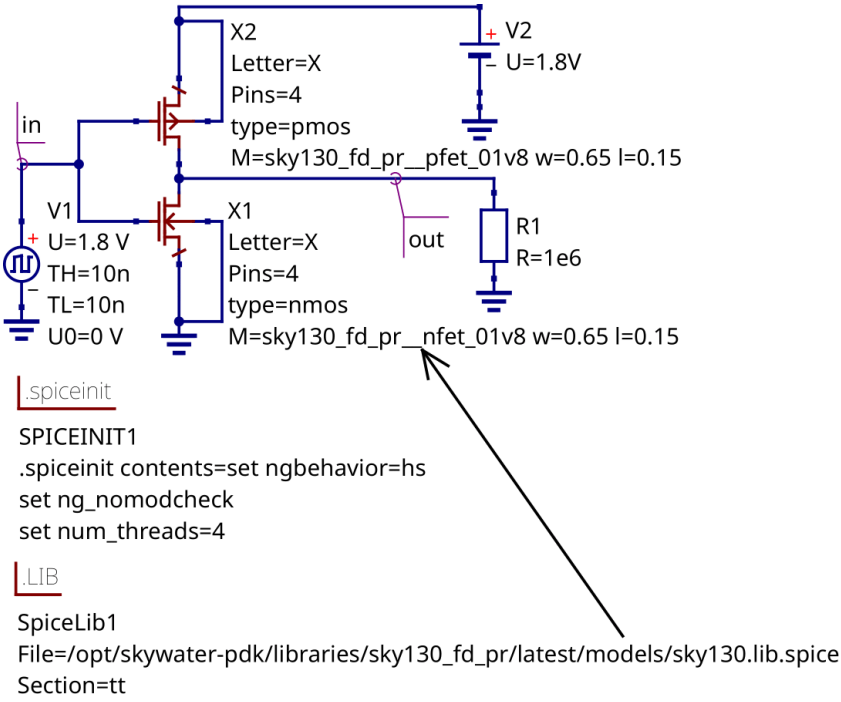

Fig. 5. CMOS IC design using Skywater PDK.

3. Compact modelling features

3.1 Verilog-A synthesizer

Verilog-A language has become an industrial standard (IEEE1364-2005) for building semiconductor devices models. The usage of the Verilog-A allows to add new semiconductor devices models without the modification of the simulation engine source.

The existing Verilog-A models could be integrated in Qucs-S using OpenVAF compiler and Nsgpice simulator link. Qucs-S provides a number of advanced features for compact modeling. The arbitrary nonlinear device could be represented as unified equation defined device (EDD). This device is described by the system of N equations representing current and charge of the device branches. The current I_k and charge Q_k of the k-th branch may depend on branch voltage V_k. Current equation for k-th branch is the following:

$$I_k = f_k(V_1, \dots, V_N, I_1, \dots, I_N) \quad (1)$$

Charge equation for k-th branch is the following:

$$Q_k = h_k(V_1, \dots, V_N, I_1 \dots, I_N) \quad (2)$$

This equation set could be automatically converted to the Verilog-A current contribution:

$$I_k^{VA} = I_k + \frac{dQ_k}{dt} \quad (3)$$

Qucs-S application contains an automated Verilog-A syn thesizer. A subcircuit containing nonlinear EDD devices and passive primitive RCL devices could be converted to a Verilog A module. The block diagram shown in the Fig. 6 explains the Verilog-A synthesizer algorithm.

Verilog-A synthesizer operates similar to SPICE netlist builder. The synthesizer allows to build compact models as subcircuit combined from primitive blocks and then debug it using the tools of

circuit simulation kernel. The passive devices and sources are described by current-voltage dependency that is converted to Verilog-A current contribution operator by the synthesizer. It is possible to synthesize such complex compact models as EKV MOSFET model [12] or Angelow GaN HEMT model [13].

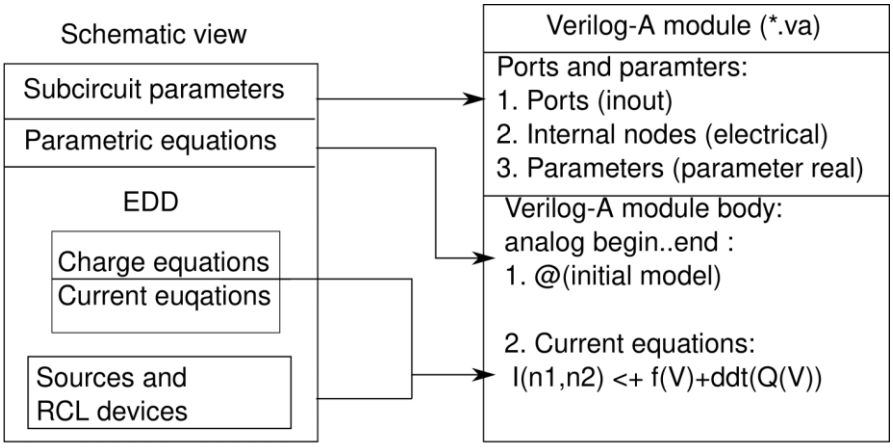

Fig. 6. The mapping of Qucs schematic and Verilog-A module.

3.2 Qucs-S, Ngspice, and OpenVAF compact model integration workflow

The Qucs-S allows to construct a new device using a special component named SPICE generic device and scripting features. This component allows to define a device with a dynamically allocated pin number and mapped to a required SPICE letter, like N for OpenVAF device. This device may be wrapped in a subcircuit. Fig. 7 illustrates this process. The tunnel diodes use Esaki model [14].

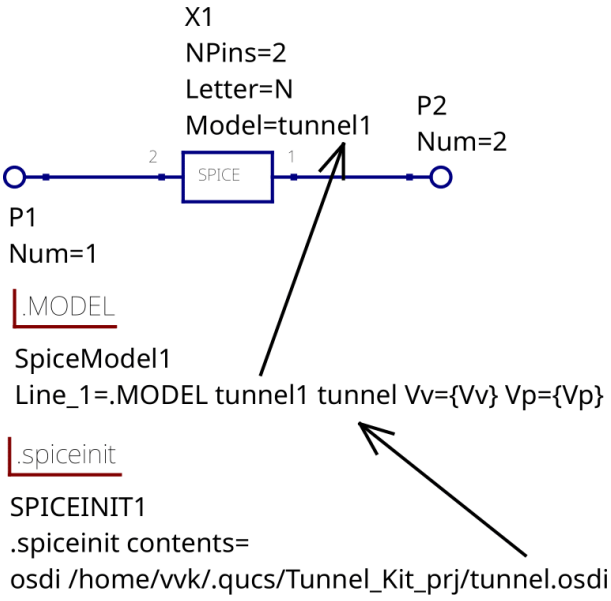

Fig. 7. Definition of subcircuit referencing OSDI module.

This device could be simulated using usual simulation procedure. The testbench schematic and IV-curve of the tunnel diode using the Verilog-A model is shown in the Fig. 8 and 9 respectively.

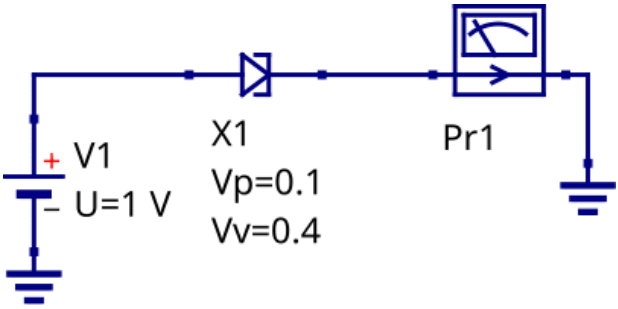

Fig. 8. Tunnel diode testbench schematic.

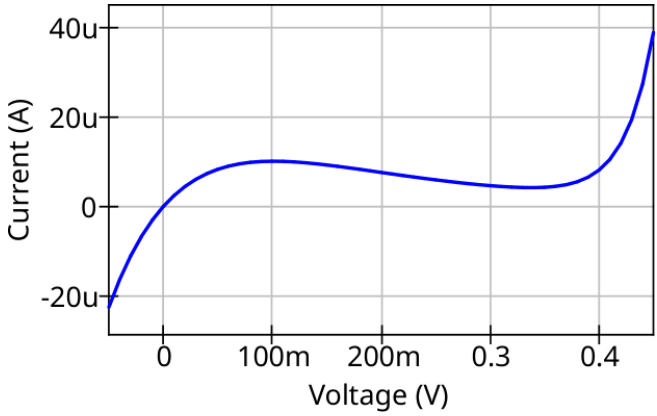

Fig. 9. Simulated tunnel diode IV-curve.

4. Semiconductor resistor modelling taking into account temperature dependency

4.1 Diffusion resistor model extraction

The simulation of the bipolar reference IC requires taking into the account temperature dependency of the diffusion resistors. The diffusion resistor SPICE model is described by the following equations:

$$R = R_{SH}\left(\frac{L}{W}\right)T_F \tag{4}$$

$$T_F = 1 + TC_1(T - T_{nom}) + TC_2(T - T_{nom})^2 \tag{5}$$

where the R_{SH} is the sheet resistance at the nominal temperature T_{nom} and TC_1, TC_2 are the temperature coefficients.

This model is fully supported by Qucs-S with Ngspice backend. The model parameters could be extracted solving this equation system numerically using the experimental data. The simulated and measured resistance temperature dependencies are shown in the Fig. 10. The Qucs-S testbench schematic and extracted SPICE model with parameters are shown in the Fig. 11.

The measurement was done using a test structure containing 12 diffusion resistors. The temperature was swept in range from-60°C to 125°C. The extraction of the R_{SH}, TC_1, TC_2 parameters was made from averaged resistance temperature dependency curve. The simulation shows a fitting of extracted model and measurement data within 1% tolerance. The extracted model was used for the design of the bipolar current reference.

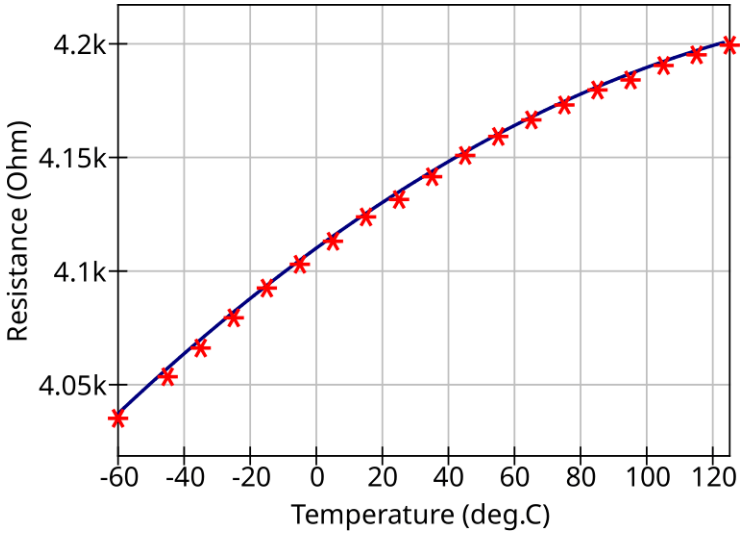

Fig. 10. Measured (stars) and simulated (solid line) temperature dependency of the diffusion resistor.

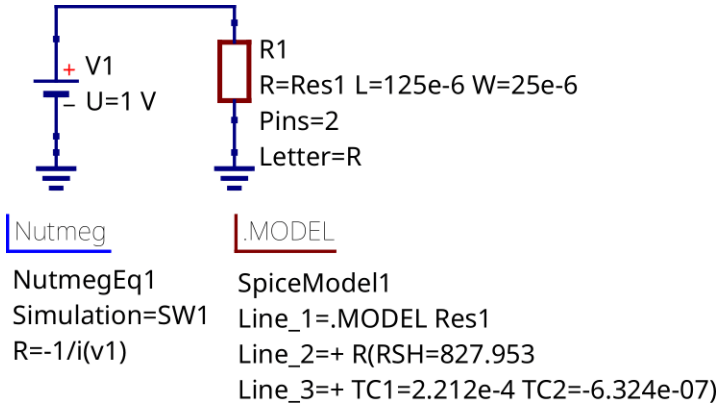

Fig. 11. Diffusion resistor testbench schematic.

4.2 Corner analysis of schematic with polysilicon resistors

Let's consider how to implement a corner analysis [15] using open-source tools. Corner analysis allows to make investigation of the worst case process parameters on the circuit performance. The testbench schematic implementing corner analysis using the Qucs-S is shown in the Fig. 12.

The library RPOLY contains three sets of polysilicon resistors models (minR, maxR, typ). The model parameters were extracted using the procedure described in the section IV-A. The extracted models should be placed in an appropriate section of the model library. The Qucs-S provides full supports of the sectioned libraries. The content of the library is shown in the listing 1.

The result of the temperature sweep using corner analysis testbench schematic is shown in the Fig. 13. The corner analysis was used for design of current reference ICs.

5. Conclusion

The open-source circuit simulation tool Qucs-S has been considered in this paper. The main advantage of this tool is that is free and open-source. The open model of the software development allows any individual or industrial user to request a new feature or propose a patch containing

improvement or bugfix. The system devices library provides a good compatibility with open microelectronic PDKs. Other advantages are the switchable simulation kernel and good support of the SPICE-3f5 standard and Verilog-A compact models using OpenVAF compiler. The QucsatorRF simulation kernel allows to use advanced RF and microwave devices models. The Verilog-A synthesizer allows to easily debug and build compact models which is especially important for academic users. The scripting and postprocessing subsystem allow use the proposed tools for model parameter extraction and new simulation construction like corner analysis. Summarizing all above we can conclude that the proposed workflow may serve as the replacement of the proprietary circuit simulation tools for educational tasks and for operation with open PDKs.

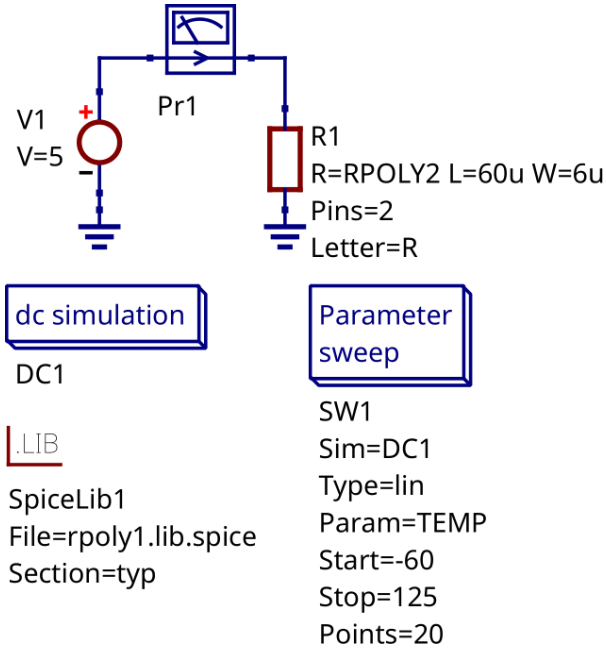

Fig. 12. Corner simulation testbench schematic.

```
* Typical corner
. LIB typ
.MODEL RPOLY2 R(RSH=1400, TC1=-0.0014, TC2=3.6E-6)
. ENDL
* min RSH corner
. LIB minR
.MODEL RPOLY2 R(RSH=1200, TC1=-0.0014, TC2=3.6E-6)
. ENDL
* max RSH corner
. LIB maxR
.MODEL RPOLY2 R(RSH=1400, TC1=-0.0014, TC2=3.6E-6)
. ENDL
```

Listing 1. The sectioned SPICE library containing the corner model.

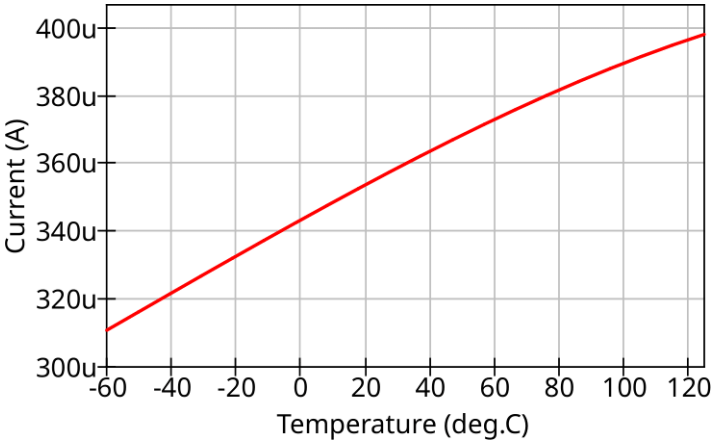

Fig. 13. Temperature dependency obtained from corner simulation.

References

[1]. A. Vladirmirescu, “Shaping the history of SPICE,” IEEE Solid State Circuits Magazine, vol. 3, no. 2, pp. 36–39, 2011. doi: 10.1109/MSSC.2011.942105.

[2]. S. Jahn and M. Brinson, “Interactive compact device modelling using Qucs equation-defined devices,” International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, vol. 21, no. 5, pp. 335–349, 2008. doi: 10.1002/jnm.676.

[3]. M. Brinson and S. Jahn, “Qucs: A GPL software package for circuit simulation, compact device modelling and circuit macromodelling from dc to rf and beyond,” International Journal of Numerical Modelling: Electronic Networks, Devices and Fields, vol. 22, no. 4, pp. 297–319, 2009. doi: 10.1002/jnm.702.

[4]. M. Brinson and D. Tomaszewski, “Advances in Qucs-S schematic capture for SPICE and Verilog-A device modelling and circuit simulation,” in 2022 29th International Conference on Mixed Design of Integrated Circuits and System (MIXDES). IEEE, 2022. doi: 10.23919/MIXDES55591.2022.9838038 pp. 27–32.

[5]. F. Lannutti, P. Nenzi, and M. Olivieri, “KLU sparse direct linear solver implementation into NGSPICE,” in Proceedings of the 19th Interna tional Conference Mixed Design of Integrated Circuits and Systems MIXDES 2012. IEEE, 2012, pp. 69–73.

[6]. M. Brinson and F. Salfelder, “Compact device modeling and simulation with Qucs/Qucs-S/Xyce modular libraries,” in 2021 28th International Conference on Mixed Design of Integrated Circuits and System. IEEE, 2021. doi: 10.23919/MIXDES52406.2021.9497545 pp. 35–40.

[7]. I. V. Chukhraev and V. E. Drach, “Parameter optimization of broadband interference-suppressing filters of hi-performance power supplies,” in 2023 5th International Youth Conference on Radio Electronics, Elec trical and Power Engineering (REEPE), vol. 5. IEEE, 2023. doi: https://doi.org/10.1109/REEPE57272.2023.10086788 pp. 1–5.

[8]. D. V. Andreev, G. G. Bondarenko, V. V. Andreev and S. A. Loskutov, "Programmable set to monitor charge state change of MIS devices under high-fields," *2022 Moscow Workshop on Electronic and Networking Technologies (MWENT)*, Moscow, Russian Federation, 2022, pp. 1-3, doi: 10.1109/MWENT55238.2022.9802396.

[9]. P. Kuthe, M. Müller, and M. Schröter, “VerilogAE: An open source Verilog-A compiler for compact model parameter extraction,” IEEE Journal of the Electron Devices Society, vol. 8, pp. 1416–1423, 2020. doi: 10.1109/JEDS.2020.3023165.

[10]. P.-C. Medinceanu and M. Enachescu, “Open Skywater130nm PDK based IP development platform: A PWM peripheral case study,” in 2024 IEEE 22nd Mediterranean Electrotechnical Conference (MELECON). IEEE, 2024. doi: 10.1109/MELECON56669.2024.10608632 pp. 856-861.

[11]. K. Herman, R. Scholz, and S. Andreev, “Reflections on the first european open source PDK by IHP-experiences after one year and future activities,” in 2024 31st International Conference on Mixed Design of Integrated Circuits and System (MIXDES). IEEE, 2024. doi: 10.23919/MIXDES62605.2024.10614043 pp. 19–22.

[12]. I. V. Chukhraev and V. E. Drach, “Investigation of factors, affecting the behaviour of Id–Vg shift in MOSFET,” in 2024 6th International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE). IEEE, 2024. doi: https://doi.org/10.1109/REEPE60449.2024.10479923 pp. 1–5.

[13]. M. E. Brinson and V. Kuznetsov, A new approach to compact semiconductor device modelling with Qucs Verilog-A analogue module synthesis. *Int. J. Numer. Model.,* vol. 29: pp. 1070–1088., 2016 doi: 10.1002/jnm.2166.

[14]. M. E. Brinson and V. Kuznetsov Extended behavioural device modelling and circuit simulation with Qucs-S. *International Journal of Electronics*, vol. *105*(3), pp. 412–425. doi:10.1080/00207217.2017.1357764

[15]. M. Lotfi and D. Zohir, “Pspice implementation of a new esaki tunnel diode macro-model,” International Journal of Smart Home, vol. 10, no. 4, pp. 153–166, 2016. doi: 10.14257/ijsh.2016.10.4.15.

[16]. G. Rappitsch, E. Seebacher, M. Kocher, and E. Stadlober, “SPICE modeling of process variation using location depth corner models,” IEEE transactions on semiconductor manufacturing, vol. 17, no. 2, pp. 201 213, 2004. doi: 10.1109/TSM.2004.826940.

Информация об авторах / Information about authors

Вадим Вадимович КУЗНЕЦОВ – кандидат технических наук, доцент кафедры Проектирование и технология электронных средств МГТУ им. Н.Э. Баумана. Сфера научных интересов: разработка САПР для схемотехнического моделирования, разработка аналоговых микросхем, воздействие электростатического разряда на полупроводниковые приборы.

Vadim Vadimovich KUZNETSOV – Cand. Sci. (Tech.), Associate Professor of the Electronic engineering department of the Bauman Moscow State Technical University. His research interests include open-source circuit simulation software development, analog ICs design, ESD robustness of microelectronic devices.

Дмитрий Владимирович АНДРЕЕВ – кандидат технических наук, доцент кафедры Проектирование и технология электронных средств МГТУ им. Н.Э. Баумана. Сфера научных интересов: разработка аналоговых микросхем, физика МДП-структур, физическое моделирование полупроводниковых приборов.

Dmitrii Vladimirovich ANDREEV – Cand. Sci. (Tech.), Associate Professor of the Electronic engineering department of the Bauman Moscow State Technical University. His research interests include analog ICs design, MOS structures physics, the physical modelling of the semiconductor devices.

Григорий Константинович ГРИШИН – студент магистратуры кафедры Проектирование и технология электронных средств МГТУ им. Н.Э. Баумана. Сфера научных интересов: свойства материалов, разработка и производство аналоговых микросхем.

Grigorii Konstantinovich GRISHIN is a Master student of the electronic engineering department of the Bauman Moscow State Technical University. His research interests include properties of materials, development and production of analog chips.