Chair for Design Automation

Technical University of Munich

March 2022 to February 2025

Contact

Technical University of Munich
School of Computation, Information and Technology
Chair for Design Automation
Prof. Dr. Robert Wille

Arcisstrasse 21
80333 Munich
Germany

cda.cit.tum.de

robert.wille@tum.de
+49 (0) 89 289 23551

Opening Remarks

Prof. Dr. Robert Wille
Chair for Design Automation
Technical University of Munich

Excellent research, excellent teaching, excellent service to our scientific community and university – and, ideally, some fun along the way. Those were the goals when, in March 2022, we started the Chair for Design Automation at the Technical University of Munich. Now, three years later, I am happy and proud to say that we delivered. In all these aspects that are at the core of work at a university, we managed to achieve numerous and outstanding results that have been widely recognized and appreciated by the community – and even provided the basis for our first start-up.

Of course, this was only possible with the tremendous support by our university, the Bavarian State Government (and particularly their support through the Bayerische Spitzenprofessur as part of the High Tech Agenda Bayern), the funding agencies who supported us (in particular the European Research Council, the European Union, and the Bundesministerium für Wirtschaft und Klimaschutz), our colleagues and partners, our students, as well as the best team someone can wish for.

For all of them, as well as for our friends and family who supported us in our passion, we created this report. It aims to summarize, in layman terms, what we worked on and what we accomplished in the past years. We hope this small "love letter" to our work conveys our enthusiasm and dedication, and gives you a glimpse into why we find what we do so fascinating.

Table of Contents

Key Figures

In March 2022, the Chair for Design Automation was founded at the Technical University of Munich. This page provides an overview of the key figures that highlight our accomplishments in the first three years.

28 Group Members
1 Professor | 3 Postdocs | 22 Doctoral Candidates | 1 Assistant | 1 Technician

9 Teaching Courses
including lectures and exercises, practical trainings, as well as seminars with 10 to 1,300 students

>50 Technical program committees served

10 Mio. Euros in funding raised

Many organized events, conferences, special sessions, tutorials, etc

4 Sophisticated Open Source Toolkits
downloaded more than 2 million times and received more than 1,000 stars on GitHub

Frequent visits from international guests

>150 Scientific papers in journals and conferences

>70 collaborations, partners, and contacts from academia and industry

1 Service Award

12 Research Awards
plus further nominations

9 External Doctoral Candidates

3 Teaching Awards

Numerous given keynotes and invited talks

Foto: MQV | Jan Greune

Research 1

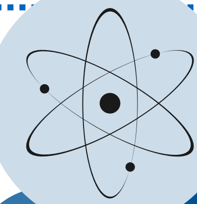
Design Automation

The Chair for Design Automation develops methods that support engineers and end users in designing complex systems and applications. This includes the *automatic* design and simulation of electronic circuits and systems, as well as future and emerging technologies such as quantum computing, microfluidics, and nanotechnologies. Additionally, we have successfully applied the methods we developed to complementary areas, such as train control systems.

The following pages provide a more in-depth summary of the research topics we are working on. We also showcase our projects, network of partners and contacts, research exchanges, and awards – and offer you a glimpse into the start-up we founded.

Circuits and Systems

Our world is dominated by electronic circuits and systems. These devices are composed of millions or even billions of components, making manual design impossible. We develop design automation methods that empower designers and engineers to tackle this complexity.



Quantum Computing

Quantum computing introduces a new paradigm that not only works with electric signals (abstracted as 0 and 1) but also utilizes quantum-mechanical properties such as superposition, entanglement, and more. We develop design automation methods to address the additional complexity resulting from this paradigm shift.

Design Automation

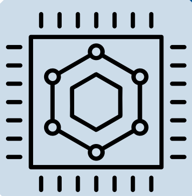
Due to the ever-increasing complexity, the design of today's (and tomorrow's) computing technology can no longer be done manually anymore. Instead, we need design automation, i.e., automatic methods and tools. We develop such design automation solutions at our chair.

Microfluidics

Microfluidics allows medical analyses, typically conducted in dedicated laboratories, to be executed on a single chip-like device. We develop design automation methods that enable the efficient realization of designs for these microfluidic devices.

Nanotechnology

Nanotechnology is an emerging field that allows operations at the atomic level, enabling ultra-fast, energy-efficient computing. We develop design automation methods that take corresponding design constraints into consideration.



Train Control Systems

Design automation can also be useful in fields beyond computing. For example, we use these methods to improve train control systems, thereby contributing to a better future and sustainable transportation.

Design Automation for Circuits

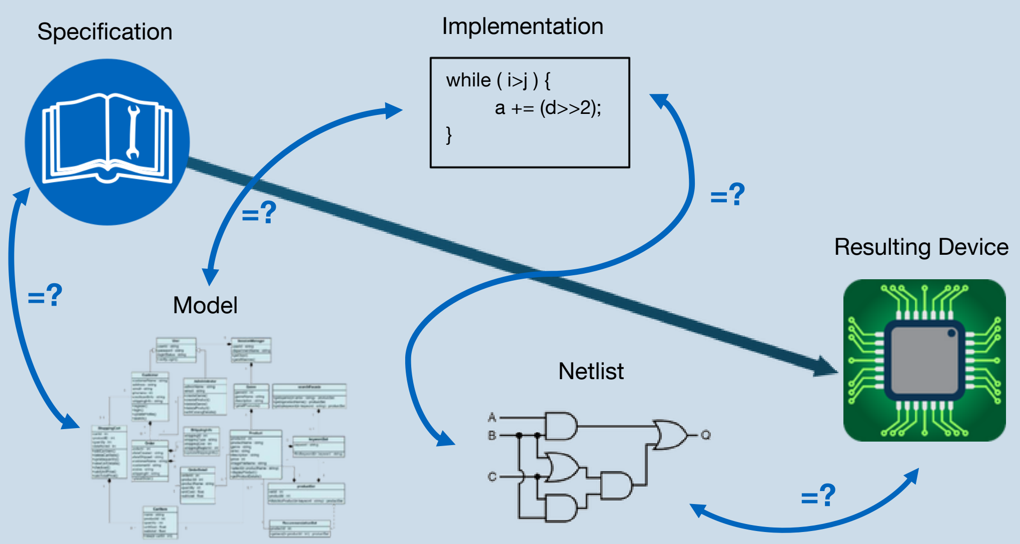
We live in a world dominated by electronic circuits: personal computers, smartphones, cars, industrial machines, and virtually every powered device. Those devices eventually became possible because, 80 years ago, electrical engineers built the first electronic circuits. Over time, they evolved the technology, made it scalable, powerful, and ubiquitous.

Current systems are composed of billions of components and are tremendously difficult to design and realize. In fact, it requires computer scientists and design automation experts to enable the utilization and penetration of electrical systems in almost all aspects of our daily life. We have programming languages, compilers, synthesis tools, verification methods, testing methods, debugging tools, and so on.

Only with these methods can we tackle the tremendous complexity of designing the next smartphone or the next AI solution.

The Chair for Design Automation covers the development of design and simulation methods for today's circuits and systems as well as for future and emerging technologies (such as quantum computing, microfluidics, nanotechnologies, and more). Besides that, we have proven to successfully apply the methods developed by us in complementary research areas (e.g., for example the design of train control systems).

Our work spans across all levels of the design stack and its corresponding abstractions (see figure).



From the initial specification over first models and implementations to resulting netlists and, eventually, the desired device. During all these steps, we constantly make sure that the obtained results are correct and as intended by testing and verifying that, e.g., a netlist is equivalent to a high-level implementation. To this end, we utilize and develop state-of-the-art methods, e.g., for logic synthesis, physical design, as well as test and verification.

Circuit Design: Simple but also complex!

At a first glance, designing today's and tomorrow's circuits and systems sounds complex – and it actually is! But the foundations are rather simple. The following three steps briefly summarize why nobody has to be afraid of circuit and system design; but also illustrate the enormous complexity which motivates design automation!

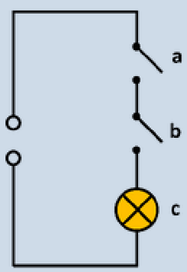
1. Computers do… well, computing tasks. Obviously, addition is one of the simpler tasks. 0+0=0. 0+1=1. 1+0=1. So far, so simple, right? But, as you probably have heard, computers only work with "power on" and "power off"; usually represented by 0 and 1, respectively. So we get kind of a problem with 1+1=2, don't we? Not really! Also our decimal number system has a limited number of digits (0 to 9). When we run out of them, we simply introduce another digit position. Computers do the same. Then, 1+1 is simply 10 (with 10 representing "our" 2).

Addition

a	b	2nd digit	1st digit
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

2. But how do computers actually do the computations? With dedicated circuits! One of the simplest circuits is the AND circuit (shown on the left-hand side). You probably ran into them in school: They have two switches (let's call them a and b) and one light (let's call it c). The light is only on (denoted as 1) if switch a *and* switch b are closed (denoted as 1), hence the name AND. If just one switch is open (denoted as 0), the light is off (denoted as 0). By this, the circuit realizes exactly what is needed to compute the 2nd digit of the table from Step 1. The first digit is realized with a similar circuit (called XOR). A simple addition can then be realized with simple circuits every kid learns in school!

AND Circuit



3. This leaves the question: How to compute more digits? Simple, just with more of these basic circuits (which are called gates). You want to do an addition with, e.g., 32-digits (or bits)? Just compose the corresponding circuits. You wanna do subtraction? Very similar concepts apply (you just need some different gates and have to compose them slightly differently). Same with multiplication, division, or whatever computation you are interested in. And that is where complexity starts to kick in. While you need just two gates for an addition with one digit, you end up with hundreds for a 32-bit addition, thousands if you add other functionality and, eventually, millions if you want to realize a full processor. No way, any human can do this manually anymore. Ever tried to solve a puzzle with 1,000 pieces? Well, think about trying one with several billions (the size of today's state-of-the-art processors)! And that is, why we need design automation (and our chair)!

You think billions of gates are complex? Well, it even gets worse: With every input you double the number of states the circuit can assume. The addition from Step 1 already has four states. If you have three (instead of two) inputs, you end up with eight states. Four inputs make 16 states, five inputs make 32 states, and so on. With 300 inputs you already have more states than atoms in the universe!

32
16
8
4
2

Logic Synthesis

Consider your smartphone, laptop, or even your car – each of these devices runs on tiny electronic circuits that bring them to life. At the heart of creating these circuits lies a fascinating process called *logic synthesis*. Think of it as the "translation" that turns high-level ideas of how a circuit should behave into an optimized blueprint that can actually be built.

Why It Matters

Today's electronic devices are incredibly complex. A single chip inside your phone might contain billions of microscopic components working together to process data, store information, or communicate wirelessly. Designing these chips is a monumental task, balancing speed, energy efficiency, and manufacturing costs. Logic synthesis is the behind-the-scenes "magic" that makes all this possible. Without it, building the next generation of tech would be like trying to construct a skyscraper without blueprints or tools.

Evolving Challenges, New Solutions

Originally, logic synthesis was developed for a relatively simple world – where most circuits followed a standard set of rules and aimed to be as small, fast, and low-power as possible. But as our needs and technologies have evolved, so have the challenges.

Modern devices integrate groundbreaking ideas like approximate computing (trading a tiny bit of accuracy for huge energy savings) or in-memory computing (processing data directly in storage to speed things up). Adapting to these advancements has made logic synthesis more sophisticated than ever. Today's researchers rely on advanced algorithms and clever mathematical tricks to navigate this complexity.

They use tools like *And-Inverter Graphs* (AIGs) and *Satisfiability* (SAT) solvers – fancy names for the frameworks that help to break down complex problems into manageable pieces. These tools ensure that the circuits are efficient, reliable, and ready for tomorrow's cutting-edge technologies.

Shaping the Future of Technology

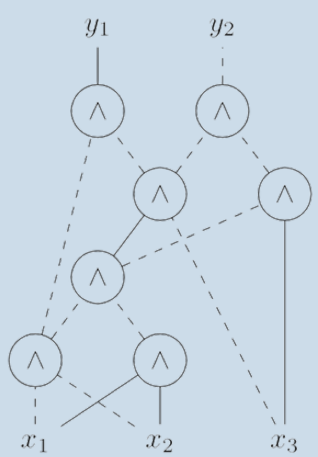
Whether it is enabling the next generation of AI chips, making your devices more energy-efficient, or preparing for futuristic breakthroughs in nanotechnology, logic synthesis plays a vital role. And while it is a complex field, our work is driven by a simple vision: to empower the designers and engineers building the technologies that shape our world.

Next time you marvel at the speed of your smartphone or the intelligence of AI, remember: somewhere in the background, logic synthesis made it all possible.

Imagine you want to add two numbers, but computers only understand 0s and 1s – like "no" and "yes". To handle this, they break the task into tiny steps using basic operations like "AND" (two times yes) and "NOT" (flip yes to no).

This *And-Inverter Graph* (AIG) represents a full adder circuit, a building block that takes three inputs and produces two outputs: the sum and the carry. The circles are AND gates, and the lines show how signals flow – dashed lines flip signals, while solid lines leave them unchanged.

Logic synthesis uses and optimizes structures like this to build efficient, reliable circuits for everything from basic arithmetic to advanced AI.



Physical Design

If logic synthesis is the art of turning a high-level circuit idea into a functional plan, *physical design* is where the rubber meets the road – transforming that plan into a layout that can be manufactured as a physical chip. It is the stage where abstract logic gets mapped to tangible geometry, balancing an intricate web of constraints like timing, power, thermal behavior, and manufacturing rules.

Bridging the Digital and Physical Worlds

Think of physical design as the architectural phase of chip creation. It is the process of deciding exactly where billions of tiny components will sit on a silicon chip and how they will connect. This includes organizing *standard cells* (the basic building blocks of a chip), designing *wiring routes* for connections, and integrating larger components like memory blocks or input/output pins. The goal? A layout that is not only functional but also optimized for performance, energy efficiency, and manufacturability.

This task has never been more challenging. As chips grow more complex and technology advances into the sub-nanometer range, physical design must juggle competing demands: making devices faster while using less power, squeezing billions of components into smaller spaces, and ensuring that designs can be reliably manufactured. Add to this the rise of new trends like *heterogeneous integration* (i.e., combining different technologies, like processors and memory, on the same chip) and you have a recipe for one of the most complex puzzles in modern engineering.

The Challenges of Chip Design

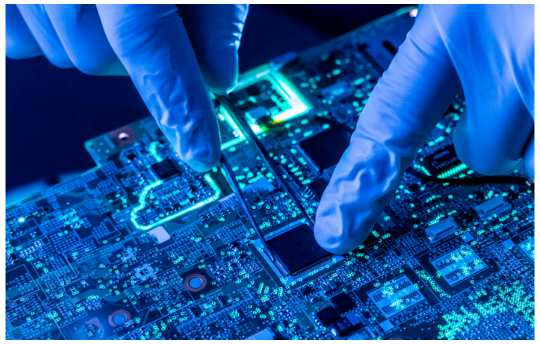
At its core, physical design is about trade-offs. For example, placing components closer together might speed up data transmission but increase heat and power consumption. Designing intricate wiring paths can minimize delays but risks congestion and manufacturing difficulties. Achieving a balance across these goals requires not only cutting-edge tools but also creative problem-solving.

The process is guided by a cascade of steps: placing components efficiently, routing the wires that connect them, and ensuring the layout adheres to strict manufacturing rules. These steps are repeated iteratively, fine-tuning the design to squeeze out every last bit of performance while ensuring that the chip remains manufacturable and reliable.

What We Do

Our chair is dedicated to advancing the tools and techniques that make physical design possible. We are tackling the toughest challenges to ensure that chips can keep up with the demands of tomorrow's technology. Here is a look at some of our work:

- **Placing Components with Precision:** Imagine arranging millions of puzzle pieces on a board – each with its own unique constraints. Advanced placement techniques do just that, using clever algorithms and machine learning to find layouts that minimize wire lengths, prevent congestion, and ensure smooth power distribution.
- **Routing the Connections:** Once components are placed, they need to be connected with microscopic wiring. This is no simple task in a world where even a tiny misstep can disrupt performance or violate manufacturing rules.
- **Exploring New Frontiers:** From supporting emerging technologies like nanotechnology and microfluidics to adapting to advanced packaging techniques like 3D integration, we are ensuring that physical design evolves alongside the technologies it supports.



Machine Learning for Design Automation

***Machine Learning* (ML), an approach in which systems learn from data to make predictions or decisions, is driving innovations in all aspects of our daily lives. Given these successes, it is only natural to explore how ML can benefit the Design Automation domain as well. However, it is crucial to determine when it makes sense to use ML and when other solutions might be more appropriate. Our chair investigates this question and applies ML techniques whenever they prove to be beneficial.**

When we want to solve a design automation problem, we typically follow one of two pathways:

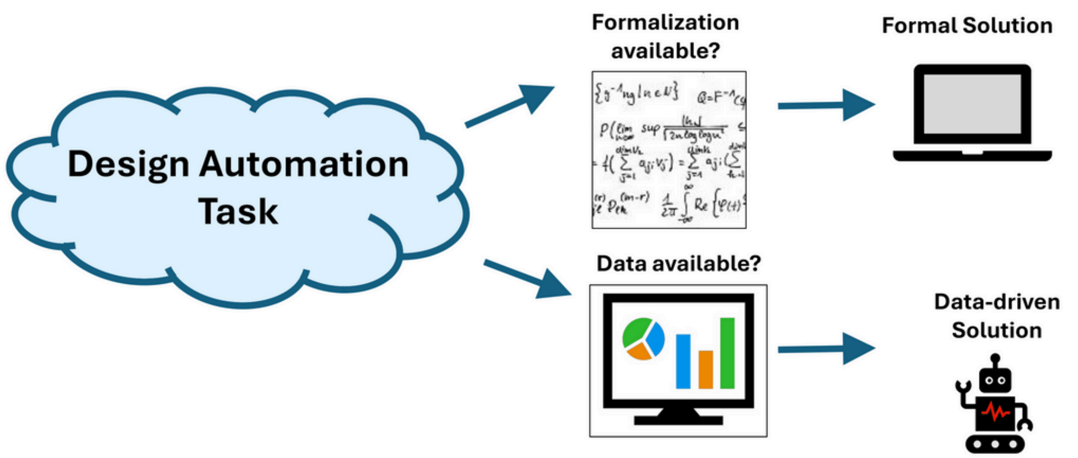
- **Formal Solutions:** If we have a well-understood problem with a clear formalization, we can engineer robust software tools to address it. Examples in design automation include logic synthesis, equivalence checking, static timing analysis, and many more. These tasks rely on solid mathematical foundations that software engineers can directly implement as specialized algorithms.
- **Data-Driven Solutions:** Many emerging design automation challenges, however, are not fully formalized yet or only partially understood. In these cases, large amounts of data may be available (e.g., design traces, test patterns, runtime logs). Here, ML-based methods can be trained to discover patterns and make predictions without requiring a perfectly specified model of the problem.

As mentioned before, designing modern electronic systems is extremely complex. Many tasks involve large datasets (e.g., design specifications, simulation data, or manufacturing results) and require both high accuracy and short turnaround times. This makes ML approaches especially appealing. For example, ML can be used for estimating design costs (e.g., performance, power, reliability) based on partial simulations or historical data. Additionally, *Reinforcement Learning* (RL) can be employed to learn how to place and route components effectively. These are just a few of the many ways ML can be leveraged in electronic design automation.

At our chair, we focus on developing and integrating ML into next-generation design automation workflows. By combining our expertise in both formal methods and machine learning, we strive to push the boundaries of design automation. We are applying these methods to:

- Traditional design automation problems such as physical design, logic synthesis, and verification.
- Emerging technologies (e.g., quantum computing, microfluidics, nanotechnologies).
- Cross-disciplinary fields, including signal processing, sensors, and more.

In this way, we aim to meet the growing complexity and demands of modern electronic systems, ensuring that new designs can be realized efficiently and reliably – much like ML is already doing in other fields.

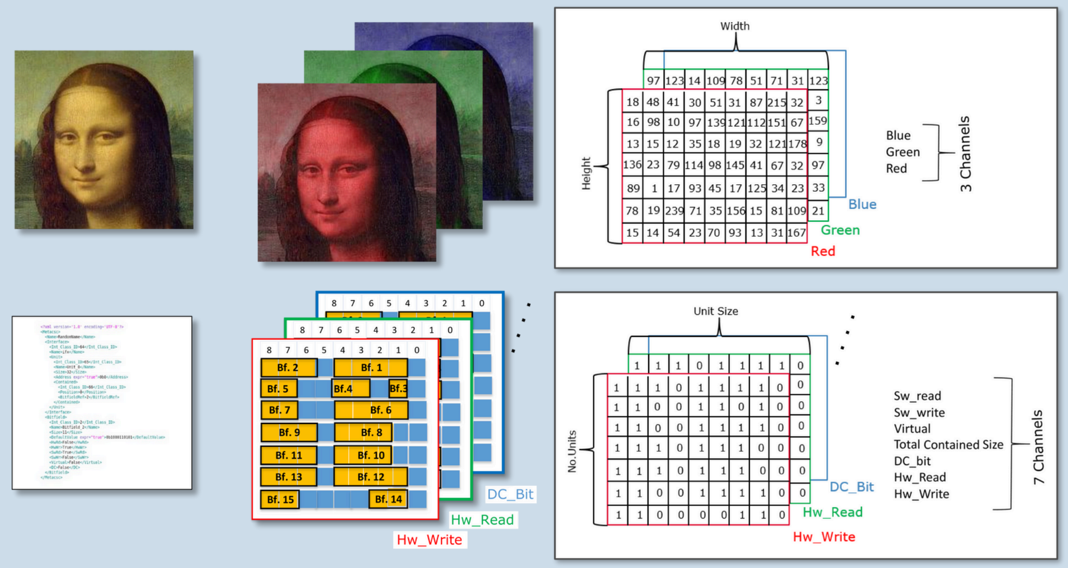


Example: Using Machine Learning for Cost Estimation

At the beginning of the design process, system designs are usually defined at high levels of abstraction to provide a variety of options for realizing a system. The choice of which design to implement depends on the corresponding costs of the different design configurations (e.g., the number of resulting components, runtime, power consumption, etc.). Unfortunately, the eventual costs are often only known once the system is fully implemented. Accordingly, methods for cost estimation are frequently applied in industrial practice. However, how to reliably estimate the costs is often not that clear. Machine Learning, particularly its application in *Computer Vision*, can help here.

In fact, Computer Vision involves analyzing and understanding images or videos for tasks like object detection and age estimation. You provide a corresponding ML tool with an image of a person, and it will predict the age of that person, often with astonishing accuracy. To this end, the image is decomposed into the three color channels (according to the RGB scheme) and represented in a fashion that can be analyzed by ML approaches such as *Convolutional Neural Networks* (CNNs) to automatically learn features from raw pixel data (as sketched in the figure below).

Now, cost estimation is very similar to age determination. But instead of providing an image of a person, we provide a system configuration, and instead of determining the age of a person, we are interested in the presumable costs of that configuration. Similarly, this system representation can be decomposed; this time into seven binary channels representing different system attributes (similar to the three color channels in an RGB image). By training ML models on these representations with known cost data, features can be learned which eventually allow for accurate and reliable cost estimation.

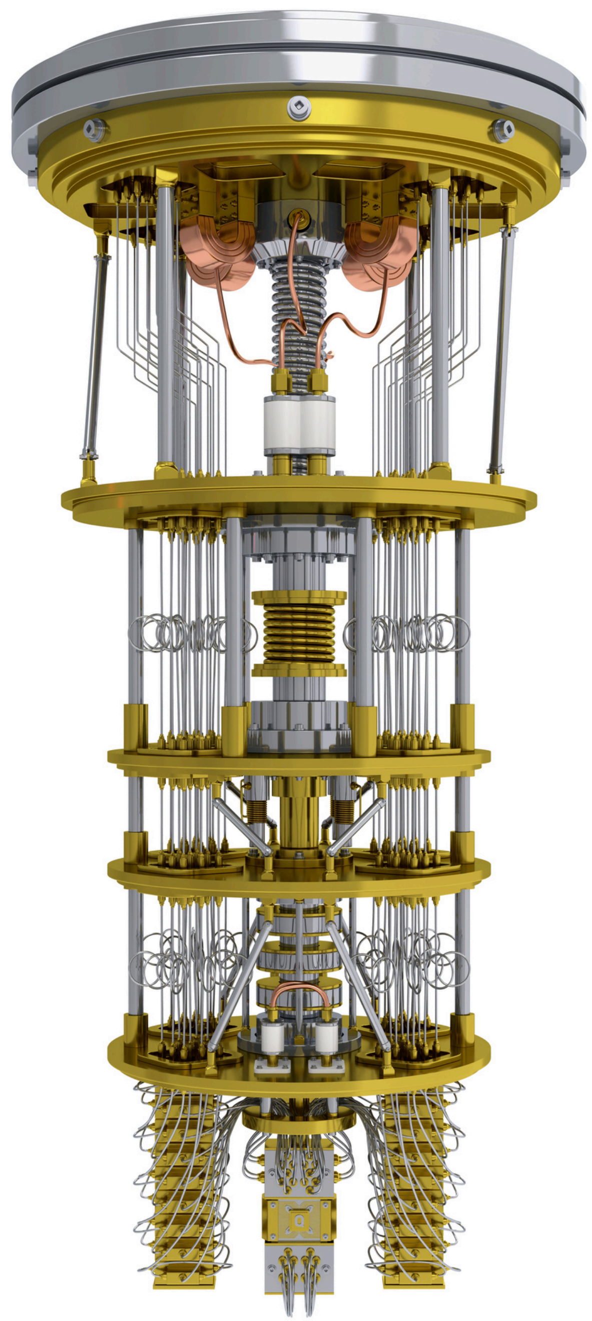


Design Automation and Software for Quantum Computing

We are currently at the dawn of a new computing age. Quantum computers – for many years just an academic dream – are becoming a reality. These machines have the potential to solve certain tasks in minutes, which conventional (super-)computers would take millennia for. Numerous quantum computing applications with a near-term perspective (e.g., in finance, chemistry, machine learning, optimization) and a long-term perspective (i.e., in cryptography, database search) are currently being investigated. Numerous research groups, established companies, and new start-ups are building increasingly powerful quantum computers – and making them accessible to everyone.

The quantum computer they are developing work in a completely different fashion than conventional devices. Rather than just dealing with 0 and 1, quantum computers work with quantum bits or qubits – which can assume any value between 0 and 1. Quantum-mechanical effects such as superposition or entanglement provide powerful capabilities, but they also lead to new challenges. Due to the radically different computational primitives, seemingly simple tasks in the design of corresponding computers and algorithms get substantially harder for quantum computing compared to conventional circuits and systems. This affects how we currently conduct design automation for quantum computing – or, more accurately put, how we do not.

Considering the design for quantum computers, we are back at square one. And this may lead to a situation where we have quantum computers but no proper (automatic) methods that aid us in using their potential!



In fact, established programming languages, compilers, and design tools, as well as test and verification schemes, do not work for quantum computers anymore. In many aspects, considering the design for quantum computers, we are back at square one. And this may lead to a situation where we have quantum computers but no proper (automatic) methods that aid us in using their potential! The Chair for Design Automation develops design automation methods and software dedicated to the realization and execution of quantum algorithms/circuits. We see ourselves as an "interface" between the stakeholders building corresponding quantum computers and the ones using them. Our research is mainly focused on (but not limited to) the topics covered on the following pages.

Computational Complexity

Many design tasks in quantum computing come with enormous complexity. Even presumably simple tasks, such as a logic simulation (which has linear complexity in the conventional domain), suddenly yield exponential complexity in quantum computing. Design automation can help here. In our work, we develop proper data structures and methods that aim to exploit redundancy in the description and make the problem more manageable.

Terminology and Formalizations

In the quantum domain, terminologies and formalizations are often unclear and ambiguous to software or design automation experts. This frequently led to situations where "wrong problems" are addressed or inappropriate models are used. Our work aims at consolidating on that and providing a proper basis that allows us to fully exploit the potential of design automation for quantum computing.

Interdisciplinarity

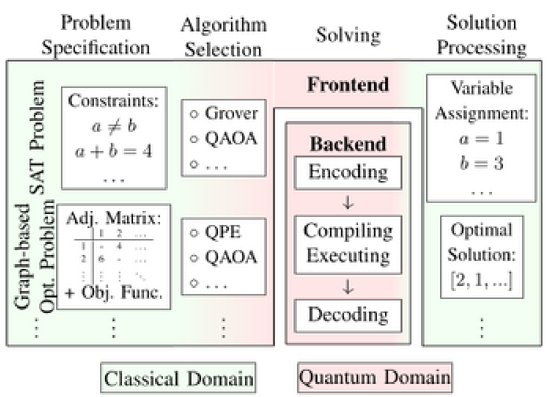
Quantum computers are developed and built by physicists. Their control requires electrical engineers. Their potential is evaluated by theorists. And, eventually, computer scientists are needed to program and realize corresponding applications. A closer interaction between these communities is key to addressing the challenges just discussed. But bridging the gap between them requires lots of resources and time to exchange with experts and stakeholders in the field. Different "languages" as well as "cultures" in the respective fields additionally impede interactions. We are working in close cooperation with all stakeholders to bring these communities together for their mutual benefit!

Supporting the End User

Tremendous progress in both quantum hardware and software has been made, especially in recent years. This has sparked interest to use quantum computing in various application domains. However, doing this is not yet as straightforward as using classical computers. Motivated by that, the Chair for Design Automation develops methodologies and software tools that support end users in using quantum computing for their application.

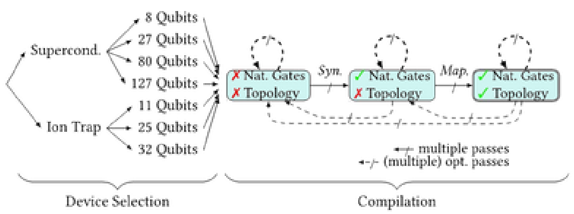
Starting with a problem from any application domain, the following steps must be followed to derive a quantum computing-based solution:
(1) A suitable quantum algorithm must be selected.
(2) The problem to be solved must be encoded as a quantum circuit, compiled for, and executed on a quantum computing machine.
(3) The resulting execution results must be decoded to extract and process the actual solution.
In many cases, this still requires dedicated quantum domain knowledge that many end users often do not have.

Motivated by this and the fact that, for classical computing, we have achieved a point where computers can be used without a deep understanding of their inner workings, we aim to provide dedicated tools that support end users of quantum computing as well. The idea is to encapsulate all tasks requiring quantum expertise (highlighted in red below) in a fully automated backend while exposing the end user only to input and output interfaces from their respective application domain (highlighted in green below).



As an example of such a tool, consider the task of selecting the most promising device and the correspondingly suitable compiler for an application. Here, you have several choices of available technologies (such as superconducting and ion traps) and corresponding devices. Additionally, for each, numerous different compilers are available as well – leading to an exploding number of different combinations. This is further complicated by the fact that a bad combination can lead to completely random results in the worst case.

To handle the resulting complexity, we developed a methodology that automatically predicts the most promising device using supervised machine learning and compiles the given quantum circuit for it using reinforcement learning. By doing so, both tasks are completely automated, and end users do not need to make these decisions themselves.



A framework for tools supporting the end user is provided as the *MQT ProblemSolver* available at https://github.com/cda-tum/mqt-problemsolver.

The tool mentioned above that automates the device selection and compilation is called *MQT Predictor* and is available at https://github.com/cda-tum/mqt-predictor.

Furthermore, we offer *MQT Bench*, a dedicated benchmarking library designed for evaluating software and design automation tools in quantum computing. With over 70,000 testcases, MQT Bench is available at https://www.cda.cit.tum.de/mqtbench/.

Quantum Circuit Simulation

Once we have a quantum algorithm (or corresponding quantum circuit), an obvious next step would be to execute it. Eventually, this should happen on the actual quantum computer. However, it often is beneficial to simulate the considered quantum circuit before on a classical computer. Since this is a tremendously complex task, efficient simulators are needed.

In fact, there are several reasons for simulation – especially during the early design stages:

- **Availability:** If quantum computing hardware is not available, simulators still allow for exploring and testing quantum applications (even if only on a smaller scale).
- **Full information:** In contrast to executions on the real machine (which only provide probabilistic measurement results), simulators offer full information on the resulting quantum states; including all of its amplitudes.
- **Error Correction:** Simulations allow for studying and developing quantum error correction methods.
- **Performance Benchmarking:** Simulations provide a baseline to compare and estimate the performance advantage of quantum computers over classical ones.

Doing the corresponding simulations on a classical computer seems simple at the first glance: Essentially quantum operations (that can be represented by unitary matrices) are applied to a given quantum state (that can be represented by a vector). Conceptually, this can be done by a series of matrix-vector multiplications. The problem: Both, the matrices and vectors exponentially grow with respect to the number of involved qubits. A severe complexity.

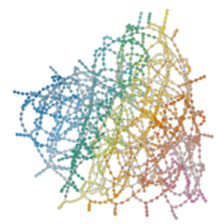
$$\begin{bmatrix} m_{11} & m_{12} & \cdots & m_{12^n} \\ m_{21} & m_{22} & \cdots & m_{22^n} \\ m_{31} & m_{32} & \cdots & m_{32^n} \\ m_{41} & m_{42} & \cdots & m_{42^n} \\ \cdots & \cdots & \cdots & \cdots \\ m_{2^n1} & m_{2^n2} & \cdots & m_{2^n2^n} \end{bmatrix} \times \begin{bmatrix} \alpha_{0\ldots000} \\ \alpha_{0\ldots001} \\ \alpha_{0\ldots010} \\ \alpha_{0\ldots011} \\ \cdots \\ \alpha_{1\ldots111} \end{bmatrix} = \begin{bmatrix} \alpha'_{0\ldots000} \\ \alpha'_{0\ldots001} \\ \alpha'_{0\ldots010} \\ \alpha'_{0\ldots011} \\ \cdots \\ \alpha'_{1\ldots11} \end{bmatrix}$$

Operation Input State Output State

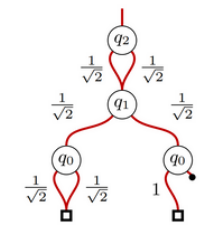
The Chair for Design Automation is actively working on tackling this complexity.

Various Simulation Methods

To simulate quantum circuits efficiently, we use dedicated data structures, such as tensor networks or decision diagrams, to represent and manipulate quantum states and operations. Additionally, we consider different types of simulations: strong and weak. Strong simulations focus on computing the entire state vector, while weak simulations aim to sample from the resulting distribution.



Tensor Network



Decision Diagram

Various Categories and Techniques

Quantum simulations can be classified as noiseless or noise-aware. Noiseless simulations produce ideal results without considering potential errors, while noise-aware simulations include error models to mimic the behavior of real quantum devices. Furthermore, there are various techniques to perform these simulations: For instance, Schrödinger-style simulations directly work with the full state vector. In contrast, *Hybrid Schrödinger-Feynman* (HSF) simulations trade memory complexity for runtime complexity by breaking the simulation into smaller subproblems.

MQT DDSIM is the simulator developed at the Chair for Design Automation, available at https://github.com/cda-tum/mqt-ddsim. It allows exploring all different kinds of simulation techniques on the basis of decision diagrams as a datastructure.

Compilation

In classical computing, compilation plays a crucial role in translating code from one programming language (the source language) into another (the target language). Typically, a compiler converts high-level, human-readable code into lower-level machine or assembly language, making it executable for a computer. This process ensures that programs can efficiently interact with different hardware. To this end, highly complex compilation chains have been developed in the past decades.

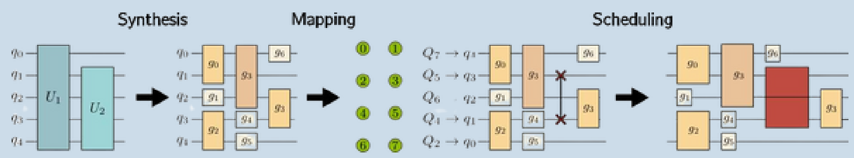
As quantum computing hardware becomes a reality, a similar infrastructure is necessary to convert high-level languages or quantum algorithms defined as mathematical matrices into a language of laser pulses, microwave pulses, or other means to control the quantum mechanical system beneath. This can be described as a three-step process as illustrated below.

In our work, we develop solutions for tasks across all three steps of quantum compilation. We provide automatic and optimized synthesis of classical reversible functions, Clifford quantum circuits, and quantum oracles (e.g., for the famous Grover algorithm). For mapping and scheduling, we provide tools for all major hardware platforms, including optimal and heuristic methods that can be integrated to other established frameworks. More precisely, we are currently supporting superconducting, neutral atom, and trapped-ion platforms. Our ecosystem also includes tools for multi-level logic compilation and a machine-learning tool for selecting the best compilation tools and their settings.

Overall, we leverage classical design automation to offer high-quality software tools for the research community. And the best of it: All tools are available in open-source!

Three Main Compilation Steps

Synthesis: Converting high-level descriptions into low-level languages similar to assembly.
Mapping: Adapting and optimizing the operations to the hardware constraints regarding connectivity.
Scheduling: Timing the resulting operations for best performance.



MQT QMAP (short for Quantum Mapper), available at github.com/cda-tum/mqt-qmap, is an open-source C++17 and Python library offering Clifford circuit synthesis and multiple different mapping strategies for superconducting and neutral atom-based hardware.
MQT IonShuttler, available at github.com/cda-tum/mqt-ion-shuttler, can be employed for ion-based devices.

MQT Qudits, available at github.com/cda-tum/mqt-qudits, is an open-source C++17 and Python library providing a self-contained compilation framework for mixed-dimensional d-level systems referred to as "qudits". This includes direct synthesis, as also the compression of binary qubit-circuits into more compact mixed-dimensional systems.

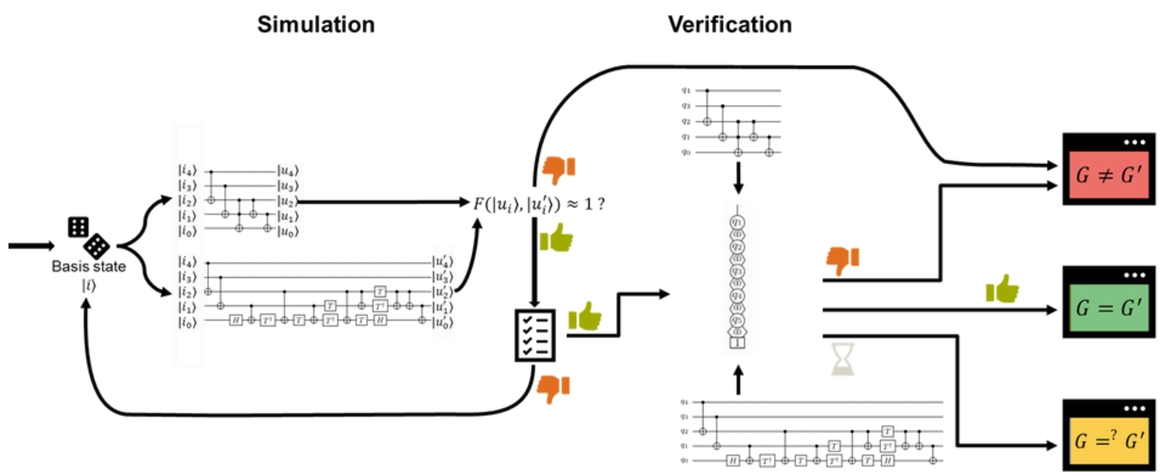
Verification

In contrast to classical computing, where errors typically manifest themselves in the form of compiler errors or program crashes, bugs in quantum programs frequently are more subtle. A quantum program might have successfully run on a quantum computer, but the obtained results may be nowhere close to what the developer expected.

Making matters worse, a study in 2021 has shown that around 40% of all bugs encountered in developing quantum computing applications stem from the tools used for their development (e.g., for compilation and optimization). This means that, when the code is not behaving as expected, there is a 4 out of 10 chance that this has nothing to do with the algorithm design itself or the programmer. Thus, days may be spent on debugging code without progress. It is crucial to provide quantum developers with automated methods to check that what is executed on the quantum computer still realizes the originally intended functionality.

The Chair for Design Automation develops such *verification* methods. Conceptually, checking the equivalence of two quantum circuits is rather straightforward: The functionality of each gate in a quantum circuit is described by a (unitary) matrix. Consequently, the functionality of a whole quantum circuit can be constructed by subsequently multiplying all individual gate matrices. Then, checking the equivalence of two circuits reduces to the comparison of the respectively obtained matrices. However, the size of the involved matrices grows exponentially with respect to the number of qubits – making it a complex problem for all but the most trivial cases.

We are addressing this problem by (1) utilizing efficient data structures for verification, (2) additionally exploiting the power of simulation, and (3) combining both (complementary) techniques into a dedicated workflow (sketched below) that utilizes the respective benefits best.

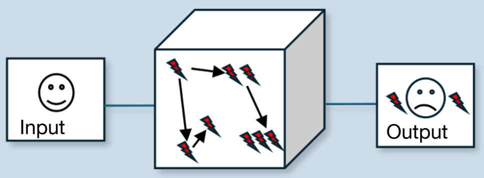


The resulting tool, *MQT QCEC* (short for Quantum Circuit Equivalence Checking), is publicly available at https://github.com/cda-tum/mqt-qcec. It implements a complex, yet fully automated, flow for checking the equivalence of two quantum circuits including simulation-based approaches as well as functional verification techniques using decision diagrams and the ZX-calculus as data structures.

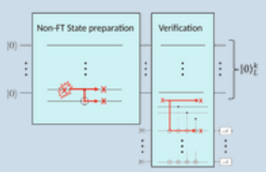
Error Correction

Due to quantum computing operating at the atomic level and the resulting sensitivity to its environment, all future large-scale quantum computers will be prone to errors. Accordingly, their application needs to be made resilient to those. Otherwise, noise would corrupt the data, and running algorithms would result in useless outcomes. Such fault-tolerant quantum computers are designed to be able to tolerate noise in the system by actively detecting errors and correcting them. However, designing fault-tolerant quantum computers requires rethinking design choices across the whole stack – from the hardware, over circuit compilation and optimization, to the software and application layer.

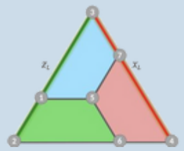
Fault-tolerance is the only way to build large-scale quantum computers that actually work. Quantum systems are particularly sensible to noise and prone to errors. When executing a quantum algorithm, errors can spread through the system and accumulate over time, thereby compromising the whole computation.



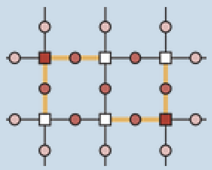
The initialization of an encoded state is an essential step in any fault-tolerant quantum algorithm. A circuit that constructs the correct state must be synthesized and this initialization circuit itself must be fault-tolerant. The size of such circuits has grave implications on the performance, so automated solutions to construct optimal circuits are important.



Quantum codes are used to protect fragile quantum information using redundancy. For instance, by encoding the state of a single qubit using multiple noisy qubits. Errors on these qubits can then be detected and corrected while protecting the precious encoded state. A code defines how the encoding works and how errors can be detected using specific measurements.



Decoders are a crucial component of a fault-tolerant quantum computer. This is a classical algorithm tasked with computing a (close to) optimal correction for errors that happening during the computation. Since they are run periodically and on huge data streams, not only must they be as accurate as possible but also highly efficient.



The resulting tool, *MQT QECC* (short for Quantum Error Correcting Codes), is publicly available at github.com/cda-tum/mqt-qecc. It offers a comprehensive and automated framework for decoding quantum LDPC codes and conducting numerical simulations. The tool supports various functionalities, including decoding quantum codes and applying error correction to quantum circuits with support for six different correction schemes. Additionally, it integrates with *Qiskit* for circuit simulation and export.

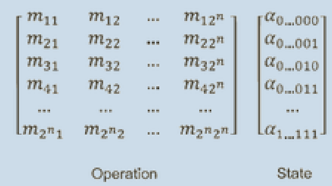
Data Structures

For all the tasks described before, data structures are key! Whenever someone aims to simulate, compile, or verify a quantum algorithm, an efficient representation, e.g., of quantum states or operations, is required. Considering that these representations often grow exponentially with respect to the number of qubits, this is not easy. Luckily, a broad spectrum of data structures is available – and is utilized as well as further improved at the Chair for Design Automation!

MQT Core, available at github.com/cda-tum/mqt-core, is an open-source C++17 and Python library, forming the backbone of the *Munich Quantum Toolkit* (MQT). It includes an intermediate representation for quantum computing, a decision diagram, and a ZX-diagram package.

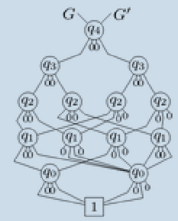
Arrays

Vectors and matrices are essential for representing quantum states and quantum operations. They can easily be implemented as arrays in classical computers. However, this method *always* requires exponential memory, limiting it to around 50 qubits, even when using supercomputers.



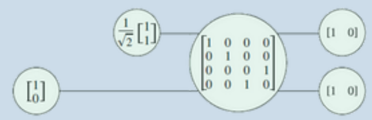
Decision Diagrams

Decision diagrams reduce exponential complexity by representing quantum states and operations as graphs, sharing identical parts to save memory. Their effectiveness varies, sometimes reducing memory needs from terabytes to megabytes, depending on the quantum state or operation.



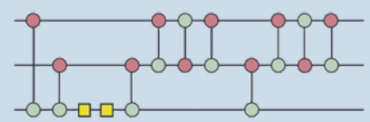
Tensor Networks

Tensor networks simplify the representation of quantum states and operations by using multi-dimensional arrays (tensors) connected according to the quantum circuit's structure. Efficiently contracting these tensors is challenging but can significantly reduce memory requirements.



ZX-Calculus

The ZX-calculus is a graphical notation for quantum circuits, using colored nodes and wires, with powerful rewrite rules for reasoning. Its flexibility, allowing transformations by bending wires, makes it a valuable intermediate language, even beyond traditional quantum circuit representations.



Activities within the Munich Quantum Valley

The *Munich Quantum Valley* (MQV) is a Bavarian initiative in which over 400 researchers work together with the primary goal of developing and operating competitive quantum computers. It promotes efficient knowledge transfer from research to industry, establishes a network with international reach, and provides educational offers for schools, universities, and companies.

The Munich Quantum Valley is supported by the Bavarian state government with funds from the Hightech Agenda Bayern. Our chair plays a vital role in this initiative, particularly in the development of software, as well as connecting to platform providers and end users of quantum computing. The following provides a brief overview of our activities within the MQV.

Munich Quantum Software Stack

In order to make quantum computing accessible to a wide range of end users from different backgrounds and fields, a comprehensive and unified software stack is needed. Such a software stack should be able to target multiple and different hardware platforms, support all kinds of potential quantum algorithms (including future algorithms that do not yet exist), provide comprehensive compilation flows, should be integrated into existing and widely deployed *High Performance Computing* (HPC) environments, and offer end user support also to users unfamiliar with quantum physics. Together with the *Leibniz Supercomputing Centre* (LRZ), the TUM-Chair of Computer Architecture and Parallel Systems, and other partners from MQV, we are developing the *Munich Quantum Software Stack* (MQSS) that takes all these requirements into account. More information: https://www.munich-quantum-valley.de/research/research-areas/mqss

Quantum Device Management Interface (QDMI)

The *Quantum Device Management Interface* (QDMI) is one of the core components of the MQSS. It enables the submission and control of gate-based quantum systems and allows software tools to automatically retrieve and adapt to changing physical characteristics and constraints of different platforms. QDMI strives to connect software and hardware developers, mediating between their competing interests, bridging technologies, and eventually providing corresponding figures of merit and constraints to be considered. QDMI is therefore the method of choice for integrating new platforms into software stacks and for software tools to query information from these platforms. More information: https://munich-quantum-software-stack.github.io/QDMI/

Group photo from the yearly meeting of the Munich Quantum Valley

Technical Exchange Meetings

Hardware providers, software developers, and eventually end users: Developing software for quantum computing is a huge interdisciplinary effort that requires connecting researchers and engineers from various disciplines. This includes physicists providing hardware, computer scientists providing software, and experts from various potential application domains (such as logistics, finance, chemistry, machine learning, optimization, etc.) providing possible use cases, among others. Each of these communities has its own language, best practices, and working style. Within the MQV, we run so-called *Technical Exchange Meetings* that frequently bring representatives from these disciplines together; not just for individual sessions, but for long-term collaboration. This ongoing interaction has enabled us to develop the aforementioned solutions. Many thanks to all our partners for this fantastic collaboration!

Impressions from the Technical Exchange Meetings.

Munich Quantum Toolkit

All developments summarized above are made available as open-source to the scientific community. This led to a collection of numerous automated, efficient, and user-friendly software tools that cater to the needs of end users, engineers, and physicists at every level of the design flow: The *Munich Quantum Toolkit* (MQT). MQT leverages extensive design automation expertise to address challenges across the entire quantum software stack.

It provides high-level support for end users in realizing their applications, efficient methods for classical simulation, compilation, and verification of quantum circuits, tools for quantum error correction, and support for physical design. These tools are underpinned by advanced data structures like decision diagrams and the ZX-calculus, as well as core methods such as SAT encodings and solvers. All tools are available as open-source implementations on GitHub.

Highlights

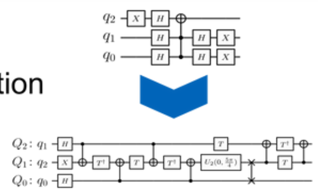
- More than 1.9 million downloads from PyPI
- More than 900 stars on GitHub
- Based on more than 100 scientific publications with over 5,000 citations
- Used in lectures at TU Munich, DTU, North-Western University, and many more

Application

- Workflow from classical problem to quantum solution
- Automated encoding, execution & decoding

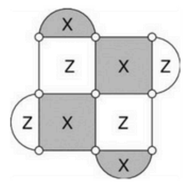
Compilation

- Automatic device selection
- Compiler optimization
- Technology-specific compilation
- Reversible synthesis



Error Correction

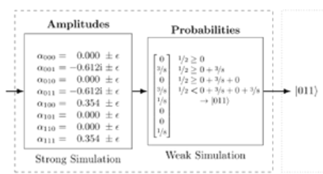
- Decoding algorithms
- Automated code construction and numerical simulations



Application → Simulation → Compilation → Verification → Error Correction → Hardware

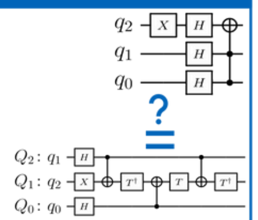
Simulation

- Classical simulation of quantum circuits based on decision diagrams
- Includes sampling, noise-aware simulation, Hybrid Schrödinger Feynman approaches, approximation strategies, expectation value computations, etc.



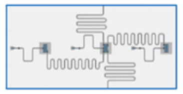
Verification

- Equivalence checking
- Verification of compilation flow results



Hardware

- Application specific physical design for superconducting platform



Data Structures & Core Methods

- Efficient data structures
- Dedicated core methods (optimal and heuristic)
- Based on C++ and Python

Decision Diagrams
Tensor Networks
ZX-Calculus
SAT/SMT Solvers
Machine Learning
Heuristics

More at:
mqt.readthedocs.io

GitHub Repository:
github.com/cda-tum

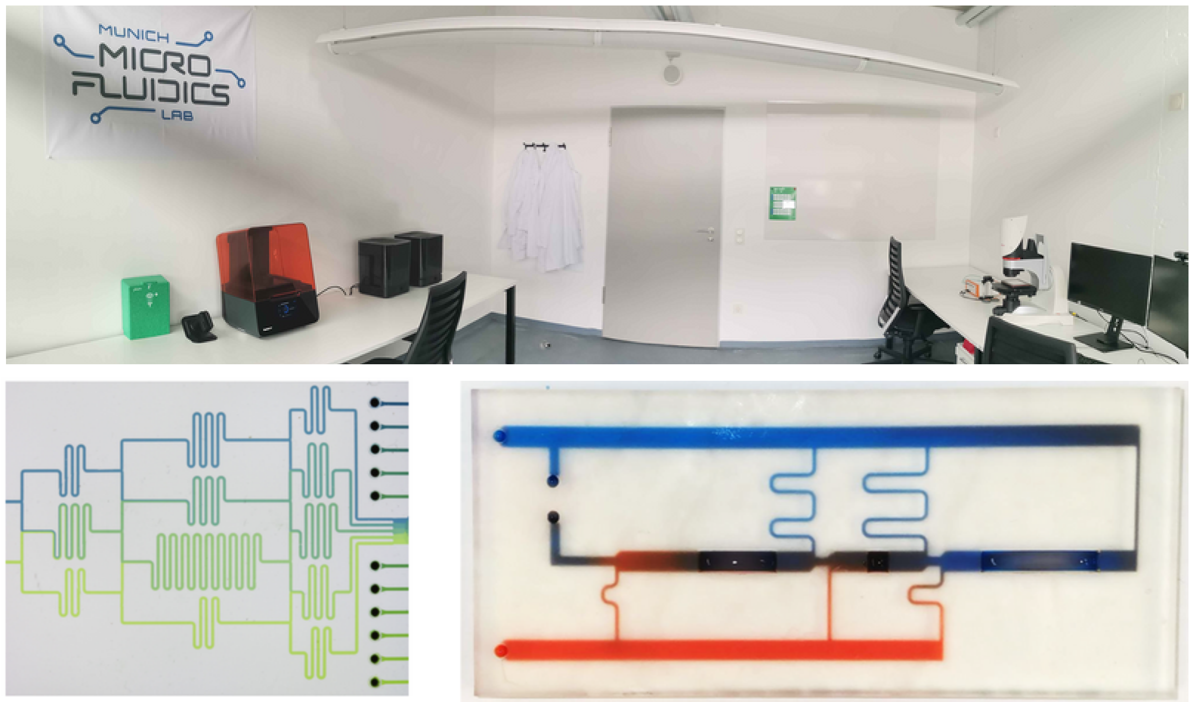
Design Automation and Simulation for Microfluidics

The COVID-19 pandemic has made it obvious to everyone: medical analyses conducted in fully-fledged laboratories require substantial amounts of time. First, the sample has to be obtained from the patient. Then, it needs to be transported to a medical lab. And, finally, the actual tests and analyses have to be conducted by trained staff using specialized lab equipment. At the same time, technology already exists that reduces sample sizes, does not require laboratory space as well as bulky equipment, and allows to run entire assays much more efficiently on so-called microfluidic biochips or labs-on-a-chip.

These devices aim to automate the involved processes including, e.g., mixing fluids (such as samples and reagents), heating them, letting them incubate, etc. In a microfluidic device, all these operations are realized on a microlevel, e.g., by networks of small channels (see figures below) through which corresponding fluids are pumped and controlled.

This allows fluids to be automatically routed through dedicated mixing chambers, heating components, or incubation paths – all realized at a microscale and, hence, in a fashion that allows for an application outside of a lab. This saves precious samples or cost-intensive chemicals and also reduces costs and overheads since operations do not have to be conducted manually anymore. Most importantly, it brings the test to the patient – truly realizing point-of-care diagnostic and treatment. A COVID-19 rapid self-test or a pregnancy test are paper-based examples of this microfluidic technology.

However, the design and layout of microfluidic devices have become considerably complex tasks. Channels must be properly dimensioned and connected, the used samples and chemicals must be injected into the chip at the right pressure, and mixing, heating, or incubation processes must be initiated at the right time. This requires dedicated expertise on a huge number of physical parameters and is mainly conducted by hand thus far.



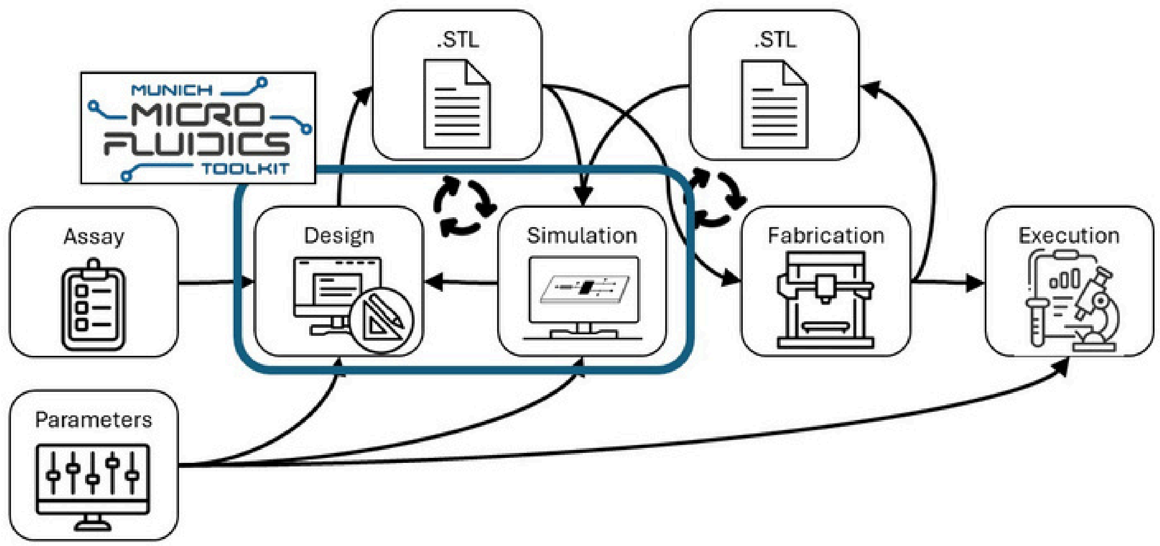
Our prototyping space enables the fabrication of microfluidic components, such as devices for fluid mixing (bottom-left) or housing structures for organ-on-chip systems (bottom-right).

Moreover, the slightest changes in the assay easily render an already existing design useless and require a re-design of the device. Accordingly, most labs-on-a-chip which successfully got established in practice are rather simple and the result of long and costly trial-and-error procedures.

The Chair for Design Automation aims to aid designers in these tasks by providing sophisticated methods for design automation and simulation. In fact, rather than manually designing corresponding devices, we are developing automatic "push-button" solutions. As input, the designer only has to provide core specifications of the desired device and the tool automatically derives a corresponding design. To this end, we utilize and develop methods similar to those from the design of electrical circuits and systems, which have been developed and optimized over the past decades.

But just automating the design process is not enough. In fact, even if a design is available, it often is not guaranteed that it works as intended. In fact, the state-of-the-art in the design of microfluidics is to fabricate the prototype, observe the functionality, and refine the design until a working device is obtained. Of course, this is very expensive in terms of time and costs.

Therefore, we propose to conduct simulations in order to validate the design, even before the first prototype is fabricated. To this end, we develop methods for simulation based on *Computational Fluid Dynamics* (CFD) as well as other abstraction levels to provide end users with efficient but also accurate estimations of the behavior of their designs.



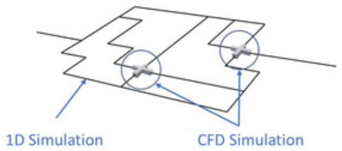
From the initial specification of an assay over first designs and simulations to a fabricated device that can be executed: The design flow for microfluidic devices as we are currently developing at our chair.

Munich Microfluidics Toolkit

In our work, we are developing design automation methods that enable the design of microfluidic devices at the push of a button, along with simulation schemes to validate those designs prior to fabrication. We aim to make the tools as accessible as possible, particularly for microfluidics experts who may not have detailed insights into design automation. Accordingly, many of our tools are provided through easily accessible GUIs and web interfaces.

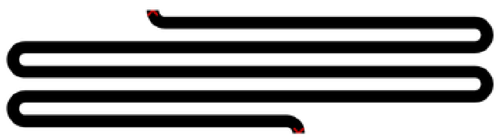
Hybrid Simulator

The *hybrid simulator* exploits the so-called 1D model to accelerate CFD simulations of microfluidic devices without loss of accuracy. This tool allows interested researchers to efficiently simulate large microfluidic networks that include components which must be simulated using a CFD method.



Meander Designer

Designers frequently draw similar designs for meander channels in a CAD program like AutoCAD. In order to overcome this manual task, we developed this online tool, which allows to automatically generate meander designs with their needs and fabrication settings.

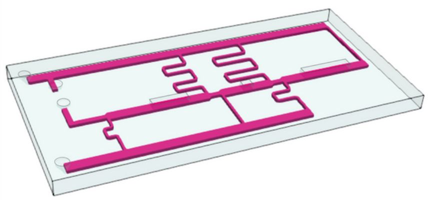


Channel Router

The *Channel Router* is a tool that lets the user design a routing layout for channel-based microfluidic devices that need to connect multiple parts in a certain way. In particular, it ensures that certain design constraints, such as the distance between channels and corner bend radius, are satisfied.

Organs-on-Chip Designer

The *Organs-on-Chip Designer* is a tool that generates an initial organs-on-chip design by considering several aspects, like the size of organ modules, the required shear stress on membranes, the dimensions and geometry of channels, pump pressures, etc. From this, a 3D geometry of the microfluidic channel network for subsequent simulations or the desired device, including the chip specifications for fabrication, can be generated.

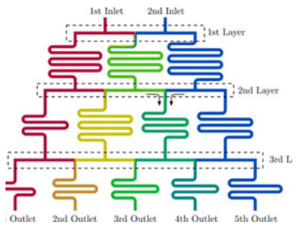


Highlights

- Combining interdisciplinary expertise among fields such as biology, chemistry, engineering, physics, and computer science
- Developing tools for diverse frontier applications, ranging from organs-on-chip and food science to fuel cells
- Tools that are freely available via web interfaces and easy-to-use for the microfluidic community

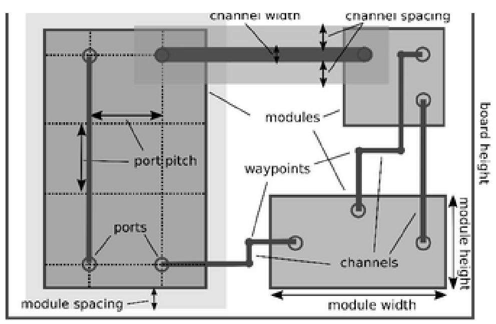
Gradient Generator

The *Gradient Generator* allows users to automatically create designs for tree-shaped concentration gradient generators. Such devices are used to mix two fluids with different concentration values.



ISO Designer

The *ISO Designer* is a tool that validates and generates microfluidic chip designs following the ISO 22916 standard.



More at:
www.cda.cit.tum.de/research/microfluidics/munich-microfluidics-toolkit

GitHub Repository:
github.com/cda-tum

Design Automation for Nanotechnology

Imagine a computer so small and efficient it could fit on a pin head while consuming almost no energy. This is not science fiction – it is the promise of computational nanotechnology. By designing circuits at the atomic level, researchers are unlocking the potential for ultra-fast, energy-efficient computing that could revolutionize everything from AI to space exploration.

But here is the catch: creating these circuits is not as simple as shrinking today's technology. At this scale, the rules change. Wires, gates, and even the way circuits are powered and synchronized all behave differently. That is where design automation for nanotechnology comes in – developing the tools and techniques needed to turn big ideas into tiny realities.

What is different at the nanoscale?

In conventional circuit design, wires are just connections – they are cheap, reliable, and do not take up much space. However, in nanotechnology, wires are made from the same materials as the gates they connect, meaning they share space and can slow things down. Similarly, clocking (the way circuits are synchronized) becomes much more complicated because signals do not just move from point A to B – they need to be carefully guided, often in a pipeline-like fashion.

For example, in new classes of nanotechnologies like *Quantum-dot Cellular Automata* (QCA) or *Silicon Dangling Bond* (SiDB) logic, even the simplest circuit layouts must consider how the information will flow and synchronize across these unconventional materials. The challenge is balancing these unique constraints while still building circuits that work reliably and efficiently.

How do we tackle these challenges?

Our chair is at the forefront of computational nanotechnology, working to make these futuristic designs practical. Here is how we are solving the nanoscale puzzle:

- **Redefining physical design:** At the nanoscale, the rules of physical design completely change – traditional algorithms no longer apply. Instead of focusing on a single technology, we develop technology-independent algorithms that work at an abstract level, making them adaptable to a wide range of nanotechnologies, from *Quantum-dot Cellular Automata* (QCA) to *Silicon Dangling Bonds* (SiDB). This flexibility ensures our tools remain useful as new technologies emerge.
- **Solving synchronization problems:** At the nanoscale, clocking is crucial but tricky. We develop methods to use clock signals more creatively, even letting them stall information when needed. This helps manage the flow of data more effectively, balancing speed and area requirements.
- **Physical simulation at scale:** Before manufacturing, every layout must be simulated to ensure it behaves as expected under quantum effects. Our simulators can analyze circuits many thousand times faster than traditional tools, helping researchers iterate quickly and accurately.

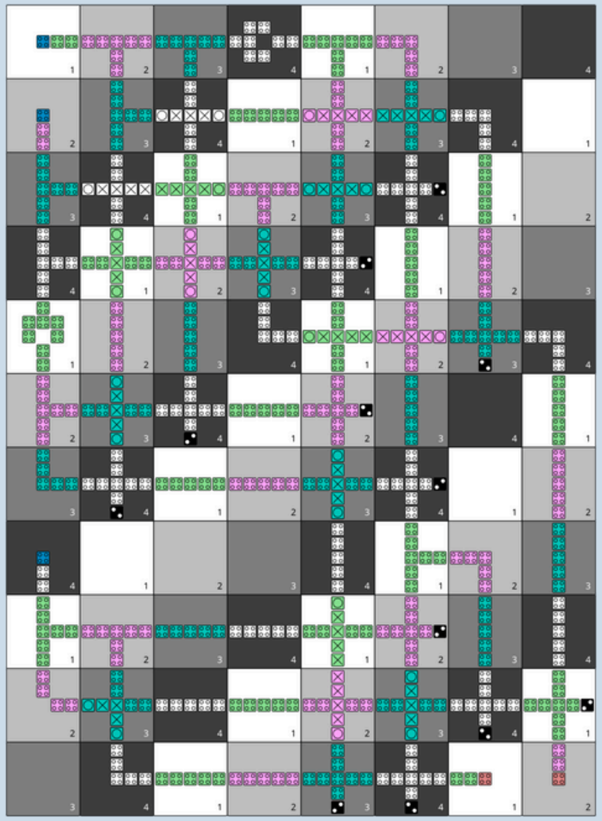
The Future of Computing

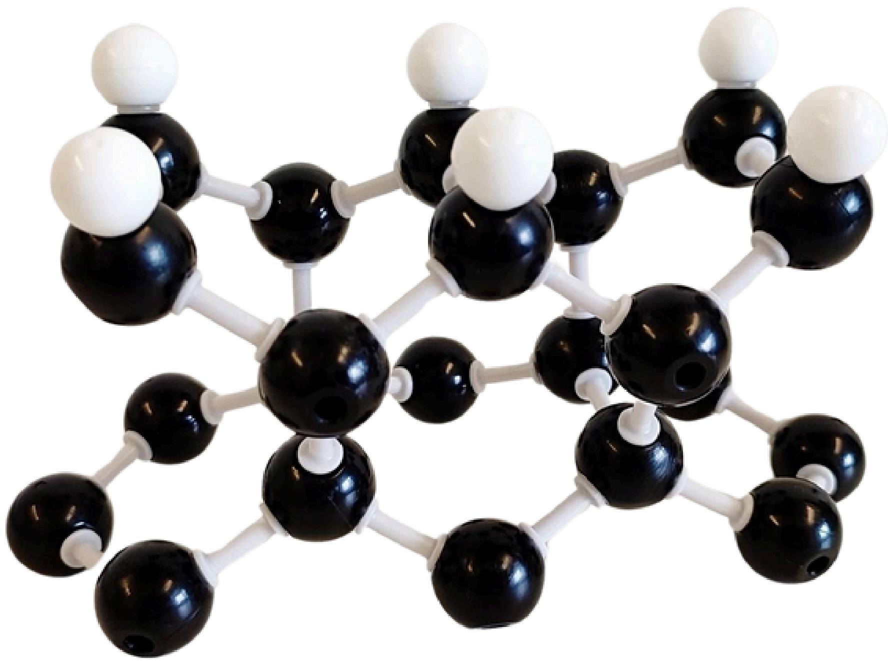
Computational nanotechnology is more than just a step forward – it is a leap into a new era of computing. By harnessing the unique properties of materials at the nanoscale, we are paving the way for devices that are smaller, faster, and greener than anything we have seen before. Design automation is the key to unlocking this potential, turning groundbreaking ideas into real-world designs. And while the challenges are immense, the possibilities are even greater. Together, we are building the foundation for the computers of tomorrow – atom by atom.

Remember the full adder from p. 14, where we saw it as an *And-Inverter Graph* (AIG)? Here is how it looks when mapped to the QCA nanotechnology. Unlike the AIG, which simplifies the circuit into logical steps, this layout must follow strict physical constraints unique to nanotech. For example, we have to carefully control the direction of data flow and ensure all parts of the circuit stay perfectly synchronized, as the circuit behaves more like a carefully choreographed dance than a simple electrical signal flow.

In this QCA layout, each small square represents a tile and each tile computes a tiny part of the overall function – like determining whether two inputs are both yes, or flipping a signal from yes to no (like in the AIG). These tiles do not work in isolation; they rely on each other, and their precise arrangement is critical. When the tiles are assembled just right, as they are here, the full adder's functionality emerges.

This transformation from an abstract graph to a physical layout showcases the challenges of working at the nanoscale and how clever design automation tools can bridge the gap, turning theoretical ideas into functional, real-world circuits.





Munich Nanotechnology Toolkit

All the research efforts described above have been combined into a powerful open-source framework for the scientific community: The ***Munich Nanotech Toolkit*** **(MNT). This toolkit offers a suite of automated, efficient, and user-friendly software tools designed to meet the needs of researchers, engineers, and physicists working at every stage of the nanotechnology design flow.**

MNT leverages cutting-edge design automation techniques to address the unique challenges of nanotech circuit design. It supports technology-independent layout generation, logic synthesis under nanotech-specific constraints, clocking and synchronization, physical simulation, and benchmarking. These tools are built on robust foundations such as satisfiability engines, machine learning models, and advanced optimization methods.

Available as open-source implementations on GitHub, MNT is designed to accelerate research and development in computational nanotechnology, enabling reproducibility and fostering collaboration across the community.

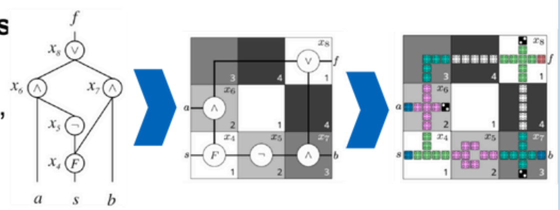
MUNICH NANOTECH TOOLKIT

Highlights

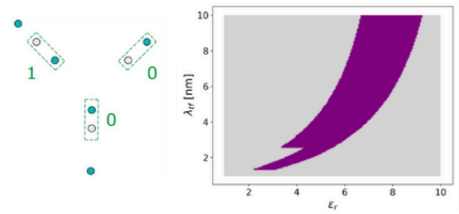
- More than 40,000 downloads from PyPI
- More than 100 stars on GitHub
- Based on more than 25 scientific publications

Physical Design

- Obtainment of **circuit layouts** from logic descriptions that involves **placement**, **routing**, clocking, timing, and technology mapping.



Simulation

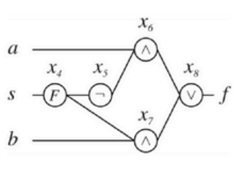


- **Ground-state** simulation.
- **Temperature** and **defect** influence regions.
- Fast **operational domain** computation.

Logic Synthesis → Physical Design → Verification → Simulation → Tool Support

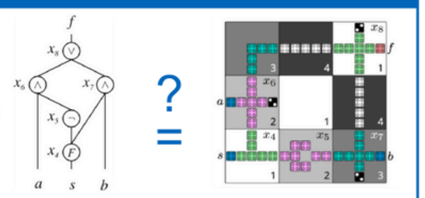
Logic Synthesis

- From RTL descriptions to technology-specific **logic network** representations.
- Unconventional **synthesis**, novel **cost functions**, and constraints.



Verification

- Layout **validation** via **design-rule** checking.
- **Equivalence** checking of **circuit layouts**.



Tool Support

- **File format** and **plug-in** support for a variety of **domain-specific tools**.

Data Structures & Core Methods

- Efficient **data structures**.
- Dedicated **core methods** (optimal and heuristic).
- Based on **C++** and **Python**.

Logic Networks · Machine Learning · SAT/SMT Solvers · Efficient Heuristics

More at:
www.cda.cit.tum.de/research/nanotech/mnt

GitHub Repository:
github.com/cda-tum

Design Automation for the European Train Control System

Railway transportation plays a vital role in future and sustainable transportation. Unlike road vehicles, trains have long braking distances. Hence, operation on sight is not feasible for praticality and safety reasons. Instead, train control systems are implemented to prevent trains from crashing into each other. In Europe, the *European Train Control System* (ETCS) has been specified as a unified control system across international borders. Similar systems have been introduced within North America (*Positive Train Control*, PTC) and China (*Chinese Train Control System*, CTCS) and even for metro systems (*Communication Based Train Control*, CBTC). While they differ in some details, the main concepts coincide. Hence, research in this field can easily be applied to most of the railway infrastructure worldwide.

Classically, railway networks are divided into fixed blocks (e.g., of length 1km). Only one train is allowed to occupy a block at any given point in time. *Trackside Train Detection* (TTD) hardware (e.g., axle counters) are needed to detect whether a particular block ist free.

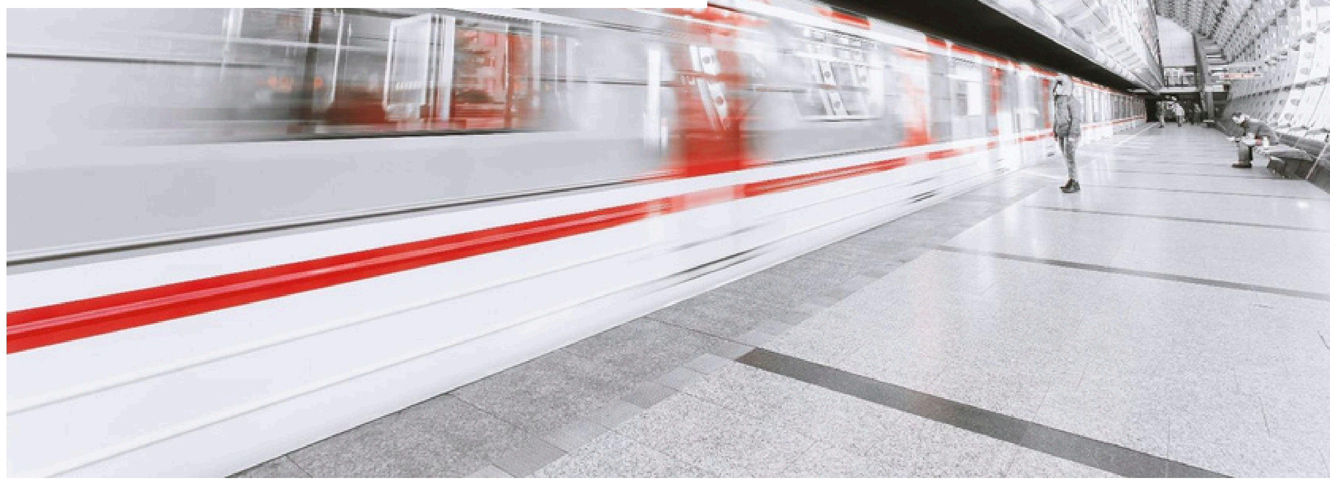
Modern trains, however, detect their exact position themselves. To some extent, this allows flexibly separating existing blocks into smaller blocks purely virtual without the need for additional hardware (*Hybrid Train Detection*). This enables trains to follow each other more closely, even in existing infrastructure. The arising smaller blocks are also called *Virtual Subsections* (VSS).

However, the question arises of where these virtual subsections should ideally be placed. Currently, planning railway tracks and their control system involves mainly manual labor based on experience and general capacity measures. It is unclear which layout enables the best operational outcome with respect to specific schedule requests.

In our group, we shed light on this and develop design automation methods that aid designers of corresponding railway networks and train schedules. Initial solutions have already been developed utilizing known design automation methods such as SAT, A*, and *Mixed Integer Linear Programming* (MILP) to generate optimal block layouts based on specific timetable requirements.

At the same time, we are developing more efficient and detailed design automation methods and ensure compatibility with digital file formats used in practice.

Our research on this topic will eventually result in a toolkit that will contain state-of-the-art methods for railway infrastructure block layout generation (see also next page). We believe that the planning process can highly benefit from such methods and resulting toolkits will be essential to optimize railway infrastructure for specific operational needs.



Motivation

Planning of railway infrastructure is mainly done by hand.

Modern train control systems allow for a greater degree of freedom.

Design automation methods are a great means to utilize this flexibility and cope with the corresponding complexity.

Design Automation

Various initial solutions have been developed utilizing known design automation methods such as SAT, A*, and Integer Programming.

Current research focuses on developing more efficient methods, optimization pipelines, and heuristics.

Compatibility

Compatibility with common data types (e.g., railML and PlanPro XML) will be implemented.

Sensible benchmark sets are to be defined.

Vision

Design automation methods are widely used during the planning process of railway infrastructure projects.

State-of-the-art solution methods are integrated into the open-source Munich Train Control Toolkit.

Munich Train Control Toolkit

The research described above is being implemented into the *Munich Train Control Toolkit* (MTCT). The functionality includes generation of optimal signaling layouts based on ETCS Level 2 Hybrid Train Detection as well as classical routing tasks on a modern Moving Block system. Most methods are based on *Mixed Integer Linear Programming* (MILP) models and academic research papers.

We aim to make the tools as accessible as possible. It is available open-source on GitHub to enable community usage.

At the same time, the toolkit is currently at an early stage and still under active development. It can be included in any C++ project.

To use the toolkit, a valid Gurobi license is needed. Academic users can obtain such a license free of charge (see also the information at www.gurobi.com/solutions/licensing/), so that the toolkit can be used without additional costs for academic research purposes.

More details on installation and requirements can be found on GitHub.

Highlights

- ✔ Open-source and freely available
- ✔ Applying design automation methods to optimal signaling layouts
- ✔ Research-driven development based on scientific papers

External Data Sources

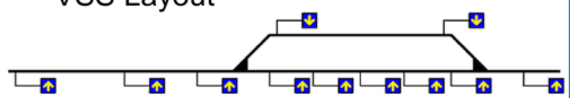
- Infrastructure
- Timetable
- Danger Points
- etc.

Solution Methods

- Exact Methods (SAT, A*, MILP)
- Iterative Optimization Techniques
- Approximation Algorithms / Heuristics
- Pre- / Postprocessing

Output

- VSS Layout



- Precise Routing

Input Data → Design Tasks → Solution Methods → Output

Design Tasks

VSS Generation: Graph/Network, Timetable → Place VSS → Minimal VSS Layout

Timetable Optimization: Graph/Network, Trains, Routes → Place VSS → Optimized Timetable

Capacity Optimization: Graph/ Network, Timetable, Trains, Routes → Place VSS → Maximal Capacity of (additional freight) trains

Abstract Data Structure

- Graph/ Network
- Trains
- Timetable
- Based on C++

Levels of Detail

- Basic Properties
- Train Dynamics
- Braking Curves
- Interlockings
- Robustness
- Delays

More at:
www.cda.cit.tum.de/research/etcs

GitHub Repository:
github.com/cda-tum

Projects

Excellent research is only possible with proper support. To this end, research projects are key. They not only provide the necessary funding but are also set up in collaborative schemes, fostering important interdisciplinary exchanges. Our chair is fortunate to have several research projects. This support enables the broad and top-notch research summarized in the previous pages. Many thanks to all funding agencies and our partners for this invaluable support!

Name: Bayerische Spitzenprofessur
Duration: March 2022 – February 2027
Funding: approx. 5 Million Euro
Provided by: Bayerisches Staatsministerium für Wissenschaft und Kunst

With the *Spitzenprofessurenprogramm* (SPP) as part of the Hightech Agenda Bayern, the Bavarian Ministry of Science promotes research at the highest level. A successful appointment within the framework of the program at a state Bavarian university is endowed with up to five million euros over a period of five years, in addition to the personnel position. This creates excellent conditions for top-level research by distinguished experts from Germany and abroad. The Chair for Design Automation is the first to be supported with this funding.

Name: Design Automation for Quantum Computing
Duration: July 2021 – June 2026
Funding: approx. 2 Million Euro
Provided by: European Research Council

The *European Research Council* (ERC) funds pioneering research across all disciplines, aiming to support innovative and high-risk projects with the potential to drive major scientific breakthroughs. Our ERC-funded project aims to develop novel methods for the simulation, compilation, and verification of quantum circuits, bridging the gap between the design automation and quantum computing communities. This way, we contribute to a stronger foundation for future advancements in quantum computing technology.

Name: Munich Quantum Valley
Duration: October 2021 – September 2026
Funding (for the chair) approx. 1.4 Million Euro
Provided by: Bayerische Staatsregierung

The *Munich Quantum Valley* (MQV) is a Bavarian initiative that unites over 400 researchers with the primary goal of developing and operating competitive quantum computers. Funded by the Bavarian state government through the Hightech Agenda Bayern, our chair plays a leading role in this initiative. We focus on software development and establishing connections with platform providers and end users of quantum computing.

Name: MILLENION
Duration: March 2023 – August 2026
Funding (for the chair): approx. 730k Euro
Provided by: European Union

MILLENION is a European partnership of 14 organizations to build and commercialize a scalable trapped-ion quantum computer with up to 1,000 qubits. Within the project, we develop new compilation methods and their integration into an HPC infrastructure to support the developed hardware.

Name: QuaST
Duration: January 2022 – December 2024
Funding (for the chair): approx. 280k Euro
Provided by: Bundesministerium für Wirtschaft und Klimaschutz

QuaST aims to provide users with the necessary software to solve complex optimization problems on current and future quantum computers. The project focuses on comprehensive software support for compiling, executing, and evaluating quantum algorithms.

Name: NeQST
Duration: 2022-2025
Funding (for the chair): approx. 560k Euro
Provided by: European Union

NeQST's vision is to leverage advances in mixed-dimensional quantum systems (i.e., systems based on qudits). This includes developing a qudit platform, automated design tools, certification methods, and practical applications. We focus on simulation and compilation for those systems.

Name: Edu4Chip
Duration: 2023-2027
Funding (for the chair): approx. 53k Euro
Provided by: European Union

Edu4Chip enhances microelectronics education to address the chip shortage crisis. The project develops courses for chip design, offering theoretical and practical training, including hands-on projects. By doing so, it aims to build a strong foundation for Europe's semiconductor plans.

Name: Generative AI for Chip Design
Duration: 2024-2025
Funding (for the chair): approx. 115k Euro
Provided by: Sony Research

Sony AI and our chair are collaborating to explore the revolutionary potential of generative AI in chip design. Generative AI promises to transform the semiconductor industry by accelerating design cycles, enhancing performance, and reducing costs. This project aims to investigate these possibilities and lay the groundwork for future, larger joint initiatives.

In addition to the projects summarized above, we are also involved in several further initiatives. These include collaborations with **Infineon Technologies**, **Fujitsu**, and others (see also our network of partners and contacts on the next page). Additionally, we are currently creating our own startup: The **Munich Quantum Software Company** (more about that on p. 50).

Network of Partners and Contacts

The research topics we work on often require interdisciplinary exchange, necessitating partners and collaborators. Besides that, great partners and contacts allow us to extend our horizons and broaden our scope. Accordingly, we are very grateful that, in the past years, we were able to build a strong network of partners and contacts with whom we collaborate, including experts from industry, academia, and research institutions. A huge thank you to all these partners for their invaluable cooperation and support.

Industry Partners

planqc
XANADU
AQT
Microsoft
IQM
IBM
FUJITSU
aws
Google
infineon
BMW
NVIDIA
parityqc
PlanQK
CLASSIQ
NEQXT
EVIDEN
SYNOPSYS
QUDORA TECHNOLOGIES
Sony AI
DATEV
intel
SIEMENS
QUANTINUUM
Quantum Silicon

Research Institutions

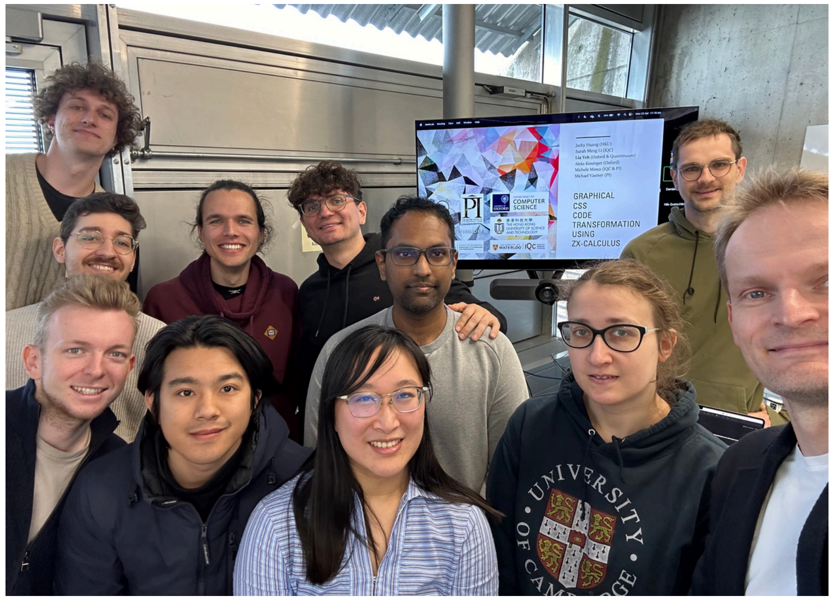
lrz Leibniz-Rechenzentrum der Bayerischen Akademie der Wissenschaften
Walther-Meißner-Institut der Bayerischen Akademie der Wissenschaften
Fraunhofer
MAX PLANCK GESELLSCHAFT
scch { software competence center hagenberg }
DLR Deutsches Zentrum für Luft- und Raumfahrt German Aerospace Center
JÜLICH Forschungszentrum
INNOVATION CENTRE DENMARK
cea leti
RIKEN

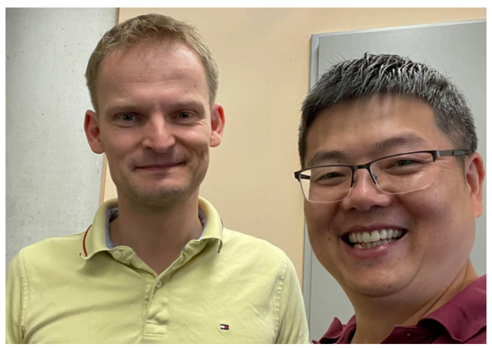
Academic Partners

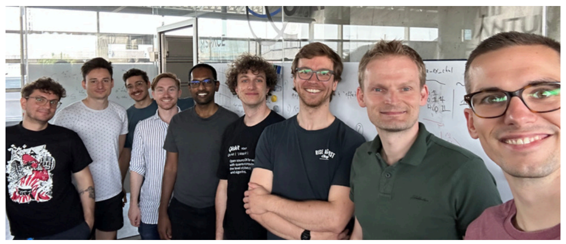
LMU Ludwig-Maximilians-Universität München
JKU Johannes Kepler University Linz
Freie Universität Berlin
RWTH AACHEN UNIVERSITY
FAU
KIT Karlsruher Institut für Technologie
TU Delft
EPFL
DTU
universität wien
Universität Bremen
OTH Ostbayerische Technische Hochschule Regensburg
universität innsbruck
UNIVERSITY OF TWENTE.
NTNU
JG|U Johannes Gutenberg Universität Mainz
Universiteit van Amsterdam
VU Vrije Universiteit Amsterdam
R 立命館大学
Università di Trento
UFMG Universidade Federal de Minas Gerais
University of Oxford
UFV Universidade Federal de Viçosa
Politecnico di Torino
Universität Stuttgart
University of Victoria
University of Cambridge
The University of Edinburgh
CHALMERS University of Technology
POSTECH Value Creating University, Pohang University of Science and Technology
The University of Utah
UCLA
University of Alberta
The University of British Columbia
UC Berkeley
جامعة نيويورك أبوظبي NYU ABU DHABI

Research Exchange

We frequently welcome guests from all over the world to our chair. During these visits, they present their work, engage in meaningful exchanges, and generate new ideas. Through these interactions, we often find synergies and create valuable collaborations. This page provides a brief selection of guests we had the honor and pleasure to host at our chair.



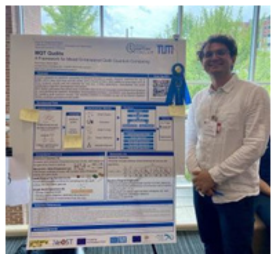




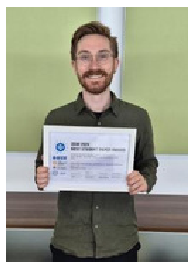
Research Awards

10-year Retrospective Most Influential Paper Award for the paper "Determining the Minimal Number of SWAP Gates for Multi-dimensional Nearest Neighbor Quantum Circuits" published at Asia and South Pacific Design Automation Conference (ASP-DAC) conference by A. Lye, R. Wille, and R. Drechsler. (January 2025)

Best Paper Award (3rd Place) for the paper "Hamiltonian-based Quantum Reinforcement Learning for Neural Combinatorial Optimization" published at the IEEE International Conference on Quantum Computing and Engineering (QCE) by G. Kruse, R. Coelho, A. Rosskopf, R. Wille, and J. Lorenz. (September 2024)



Best Poster Award for the poster on the framework "MQT Qudits" at the US Quantum Information Science Summer School 2024, hosted by Oak Ridge National Laboratory, awarded to Kevin Mato. (July 2024)

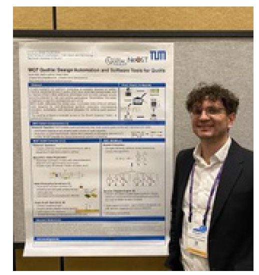


Best Student Paper Award for the paper "Stripping Quantum Decision Diagrams of their Identity" published at IEEE International Conference on Quantum Software (QSW) by A. Sander, I.-A. Florea, L. Burgholzer, and R. Wille. (July 2024)

RESEARCH AWARDS

Best Presentation Award for the presentation of the paper "Technology Mapping for Beyond-CMOS Circuitry with Unconventional Cost Functions" at the IEEE International Conference on Nanotechnology (IEEE NANO) awarded to Marcel Walter. (July 2024)

Best Poster Award for the presentation of the paper "Efficient Post-Training Augmentation for Adaptive Inference in Heterogeneous and Distributed IoT Environments" at the International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation (SAMOS) awarded to Max Sponner. (July 2024)



Unitary Fund Grant awarded to Kevin Mato for the framework "MQT Qudits", recognized as one of the most promising quantum software projects. (June 2024)

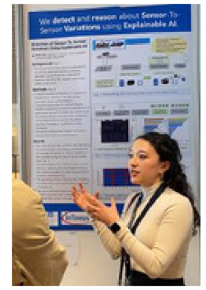
RESEARCH AWARDS

ACM SIGDA Outstanding PhD Dissertation Award given by the ACM Special Interest Group on Design Automation (SIGDA) for the dissertation "Design Automation Tools and Software for Quantum Computing" awarded to Lukas Burgholzer. (June 2024)

Heinz Zemanek Award given by the Austrian Computer Society (OCG) for the dissertation "Design Automation Tools and Software for Quantum Computing" awarded to Lukas Burgholzer. (June 2024)

Best Poster Award for the PhD Forum presentation "Design Automation Tools and Software for Quantum Computing" at the Design, Automation and Test in Europe (DATE) conference awarded to Lukas Burgholzer. (March 2024)

Best Poster Award for the presentation on "Detection of Sensor-To-Sensor Variations Using Explainable AI" by Sarah Seifi. (March 2023).



Best Student Paper Award for the paper "Automatic Implementation and Evaluation of Error-Correcting Codes for Quantum Computing: An Open-Source Framework for Quantum Error Correction" published at International Conference on VLSI Design (VLSI Design) by T. Grurl, C. Pichler, J. Fuß, and R. Wille. (January 2023)

Besides that, in the past three years, we were **nominated for the Best Paper Award** by the *Design, Automation and Test in Europe* (DATE) conference in 2022, the *Asia and South Pacific Design Automation Conference* (ASP-DAC) in 2023, and the *International Workshop on Logic Synthesis* (IWLS) in 2023. Further awards include a **JKU Early Research Achievement Award in 2022**, the **Promotio sub auspiciis Praesidentis rei publicae** in 2022, a DAC Young Fellow Best Video Award in 2022, and an **Audience Choice Award** in 2022. Finally, we also received **teaching awards** (see p. 62) and an **award for our professional service** (see p. 71).

Start-Up: The Munich Quantum Software Company

As summarized above, we have achieved numerous research results in the past three years, that have been widely recognized and appreciated. Moreover, through our exchanges with various stakeholders, we have observed great interest and demand for the methods we develop. Accordingly, we decided to take a step further. In the spirit of the Technical University of Munich considering itself "The Entrepreneurial University," we embarked on the endeavor of founding our own start-up: The Munich Quantum Software Company. This page provides an overview of what this venture is all about.

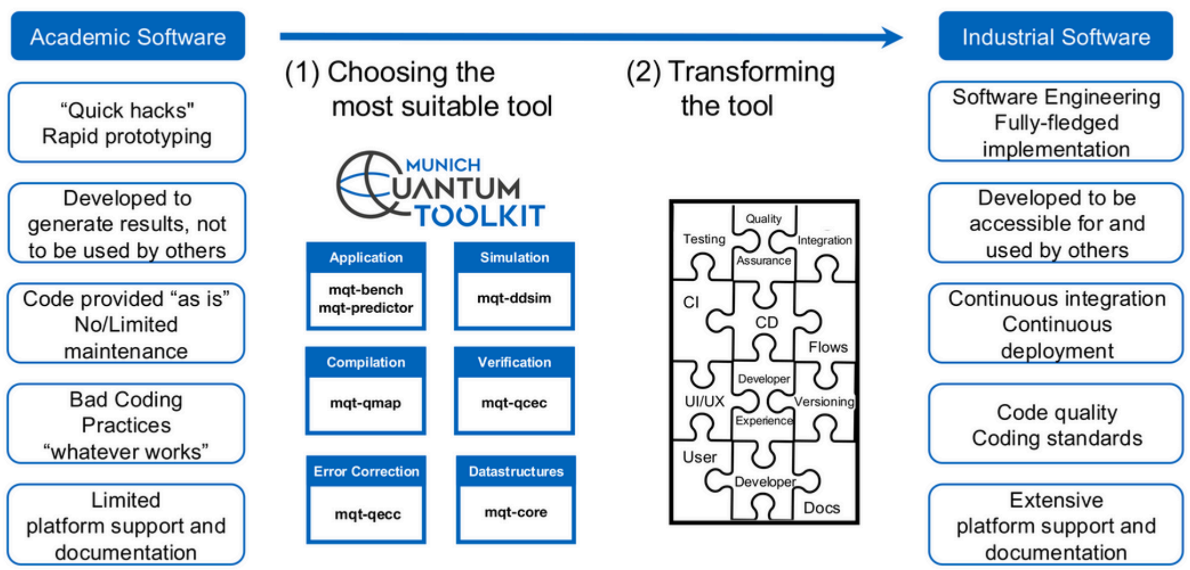
Mission

Quantum computing is becoming a reality: Superconducting, ion traps, neutral atoms,… the hardware is getting there! However, software capable of handling numerous complex design tasks is needed to connect end users to these platforms. Unfortunately, software for quantum computing is still in its infancy, and the development of quantum computing software remains challenging. The Munich Quantum Software Company aims to create industry-grade software tools and services that we already take for granted in conventional computing.

Goal: Industry-Grade Software Tools

The Munich Quantum Software Company builds on the expertise of the Chair for Design Automation, leveraging its experience in developing world-class quantum computing software to transform academic breakthroughs into robust, industry-grade solutions.

Currently, most quantum computing software developed in academia is created by individuals or small groups, primarily to demonstrate the potential of an idea. This often results in tools that are poorly documented, not widely available, and lack proper testing and maintenance – severly limiting their usability. With frameworks such as our Munich Quantum Toolkit (see p. 26), we are already aiming in that direction – adhering to high-quality standards, making use of elaborate continuous integration and deployment pipelines, as well as offering extensive documentation and broad platform support. With the company we want to realize real industrial-grade software and provide custom solutions.



The Munich Quantum Software Company aims to transform academic solutions into industrial software.

Provided Tools & Services

Munich Quantum Toolkit (MQT)
Our collection of open-source software tools for quantum computing provides solutions for design tasks across the entire quantum software stack. This already provides a strong basis that can be used to create industrial-grade software tools. The company will develop tools like those for industrial settings.

Services
The company offers dedicated services supporting end users and hardware providers. This shall ease the access to quantum computers or allow the easy evaluation of their performances. The resulting software-as-a-service solutions are provided, e.g., as web-based solutions and, hence, are easily accessible.

Software Development
Customers need a dedicated tool that is not available yet? No worries, we are ready for that! Whether it is software for quantum applications, simulators, compilers, execution tools, physical design aids, or more; the company got you covered.

Integration into Software Stacks
Need advice on creating or improving your software stack? Want to integrate new tools seamlessly? Also here, the company can help with extensive experience in developing comprehensive software stacks that connect end users to hardware platforms.

Highlights

- More than 15 years of experience in quantum software
- Tools and services rest on more than 100 scientific papers
- Based on solutions that won more than 20 prestigious awards
- Turning academic breakthroughs into industry-grade software
- Trusted by the community with 1.9M+ PyPI downloads, 900+ GitHub stars, and over 5,000 citations

Foto: MQV | Jan Greune

Teaching 2

Lectures

The Chair for Design Automation offers lectures, seminars, projects, and theses for all levels (B.Sc., M.Sc., as well as Ph.D.). We cover a broad variety of topics with a particular focus on the design of circuits and systems for conventional as well as alternative and post-CMOS computing technologies such as biochips, quantum computing, or reversible circuits. Besides that, we are constantly broadening our portfolio. We believe in research-oriented learning but also provide various topics with practically relevant applications.

Introduction into Computer Architecture

Type: Lecture and Excercise
Level: Bachelor
This lecture covers the fundamentals of computer architectures and provides a basic understanding how today's computers work and why they are capable of running complex algorithms in very short time. Practical exercises accompany the lecture to illustrate these concepts. It is a mandatory course for first-semester computer science students at the Technical University of Munich, regularly attended by over 1,000 students. The lecture is conducted in collaboration with the Chair for Computer Architecture & Parallel Systems by Prof. Martin Schulz.

Lab Course – Computer Architecture

Type: Practical Training
Level: Bachelor
In this internship, students get hands-on experience in realizing and working with processors. Students learn fundamental topics in computer architecture through lectures, tutorials, and homework, including C programming, secure programming, debugging tools, system design, circuit planning and realization using SystemC, and assembly language programming. In the second half, students work in groups on practical projects related to system design or assembly language programming, culminating in a presentation at the end of the semester.

Introduction to Emerging Computing Technologies

Type: Lecture and Excercise
Level: Bachelor
Computer technologies are evolving as Moore's Law reaches its limits. Accordingly, emerging computing technologies like quantum computing, reversible circuits, microfluidic devices (Labs-on-a-Chip), and field-coupled nanotechnologies are being explored as alternatives. This lecture provides an overview of these technologies, their paradigms, and potential applications. It also discusses how to efficiently design applications for these new technologies.

Software for Quantum Computing

Type: Lecture and Excercise
Level: Master
Quantum computers can solve tasks that would take millennia for conventional supercomputers. While many near-term (e.g., finance, chemistry) and long-term (e.g., cryptography) applications are being explored, the development of automated methods and software tools is lagging. This lecture introduces quantum computing software, covering differences from classical computing, design flows, data structures, and tools for tasks like simulation, compilation, and verification. It includes hands-on experiences with actual quantum algorithms and platforms.

Quantum Computing Software Lab

Type: Practical Training
Level: Master
This practical training complements the previous lecture by providing hands-on experience of quantum computing software. Students learn to use various software tools and design automation methods essential for developing quantum computing solutions. The training covers different frameworks and guides students in implementing selected quantum algorithms in practice, bridging the gap between theoretical concepts and real-world applications.

Design Automation and Simulation for Microfluidic Devices

Type: Lecture and Excercise
Level: Master
Microfluidics involves the precise control of fluids at the micro-scale, enabling experiments in medicine, chemistry, biology, and pharmacology on integrated, automated devices known as *Lab-on-a-Chip* (LoC). This lecture covers applications for microfluidic devices, available biochip platforms, design automation, simulation methods, and fabrication procedures. Students gain a comprehensive understanding of how to develop a microfluidic application from concept to prototype, addressing challenges like channel dimensioning, sample injection, and process timing.

Foto: MQV | Jan Greune

Machine Learning for Design Automation and Manufacturing

Type: Lecture and Excercise
Level: Master
This lecture explores how machine learning can enhance chip design, addressing the complexity and cost issues of traditional design automation toolkits. It covers the theory and application of machine learning algorithms, recent methods for design automation, and key areas in the semiconductor industry where data is abundant. Students also gain hands-on experience with state-of-the-art tools and data structures, learning to apply machine learning to optimize and improve the chip design process.

Logic Synthesis and Physical Design

Type: Lecture and Excercise
Level: Master
Modern computer chips, with billions of transistors, are designed using algorithms refined over decades. This lecture teaches techniques for automatically obtaining chip designs from specifications, covering logic synthesis, optimization, partitioning, floorplanning, placement, and routing. Students learn meta-heuristics applicable in various fields and gain hands-on experience with state-of-the-art tools like ABC, Yosys, OpenROAD, and iEDA. They also have the opportunity to participate in an international contest.

Seminar on Topics in Design Automation

Type: Seminar
Level: Bachlor/Master
In this seminar, participants discuss current topics in Design Automation. They receive an introduction to scientific literature, covering paper reading, literature research, presentation techniques, and scientific writing. By the end of the seminar, participants are equipped to independently perform all necessary steps to present a scientific topic, culminating in a review paper and an oral presentation.

Bachelor's and Master's Theses

We constantly supervise Bachelor and Master theses on a wide variety of topics with a particular focus on the design of circuits and systems for conventional as well as alternative and post-CMOS computing technologies such as quantum computing, biochips, reversible circuits, etc.

Besides that, we are constantly broadening our portfolio and are also open to exciting ideas by students. We believe in research-oriented learning, but also provide various topics with practically relevant applications. This page presents a selection of theses supervised by our chair.

Ordering of Decision Diagrams for Quantum Simulation without Reordering

Claudia Dünzinger
Bachelor, September 2024

Quantum Algorithms for Multi-Objective Optimization

Ayse Kotil
Master, May 2024

On the importance of verifying the results of quantum circuit compilation flows

Tiago Alexander Reiter
Bachelor, May 2023

Equivalence checking of quantum circuits using diff algorithms

Janis Pepe Heims
Bachelor, October 2024

Beyond Classical Boundaries: Quantum Optimization Strategies for Karp's 21 NP-Complete Problems

Petar Petrov
Master, November 2024

Explainable and Robust AI for improving People Counting Tasks using FMCW radar

Huawei Sun
Master, October 2022

Meta-Reinforcement Learning for Human Targets Tracking in heterogeneous Scene Settings

Julius Ott
Master, August 2022

Machine Learning Guided Quantum Circuit Compilation

Patrick Hopf
Master, December 2024

Depth-Optimal Synthesis of Graph State Preparation Circuits

Simone Domenici
Bachelor, July 2024

Signal Distribution Networks in Automatic QCA Standard-Cell Placement & Routing

Benjamin Hien
Master, February 2023

Optimal Design of Train Control Systems with Hybrid Train Detection using A*

Yusuke Kibakura
Bachelor, November 2024

Reinforcement Learning and Evolutionary Algorithms for Jammer Mitigation in Cognitive Radio

Simon Hofmann
Master, Juli 2022

Edu4Chip: Joint Education for Advanced Chip Design in Europe

The chip shortage crisis emphasizes the need for skilled chip design engineers. Edu4Chip aims to address this by enhancing microelectronics education through collaboration between Higher Education Institutes and industry. Our chair is a partner in this initiative.

The chip shortage crisis has highlighted the reliance on integrated circuits across all sectors and revealed a shortage of skilled chip design engineers. Despite existing microelectronics programs, the skill gap persists, necessitating enhanced education and collaboration between Higher Education Institutes and the private sector. Edu4Chip aims to develop and improve master-level courses in chip design, providing both theoretical knowledge and practical training.

This includes hands-on experience with chip design and tape-out projects, motivating students and preparing them for the European labor market. The project involves a consortium of universities, research institutes, and SMEs, focusing on strengthening existing programs, attracting diverse students, developing digital solutions, and fostering industry partnerships. By doing so, Edu4Chip seeks to close the skills gap and meet the demands of the semiconductor industry. Our chair is a proud participant in this initiative.

Coordinator:
Technical University of Munich (TUM)

Beneficiaries:
Technical University of Denmark (DTU)
Tampere University (TAU)
KTH Royal Institute of Technology
MINRES Technologies GmbH
LogiqWorks Ltd.
Institut Mines-Télécom
SyoSil ApS
Fraunhofer IIS

Associated Partners:
Infineon Technologies AG
Texas Instruments
VLSI Solution Oy

Feedback From Our Students (in German)

Professor Robert Wille ist der beste Professor des ersten Semester. Man kann seine Leidenschaft für die Themen deutlich merken und seine Vorlesungen haben eine leichte Atmosphäre. Der Professor hat keine Schwierigkeiten, uns seine Kenntnisse deutlich und vollständig beizubringen und seine Vorlesungen machen deshalb echt viel Spaß.

Herr Professor Wille erklärt die Inhalte super verständlich und sehr interessant. Aufgrund seines sichtbaren Enthusiasmus und seiner Hilfsbereitschaft gegenüber den Studenten ist es für mich fast unmöglich, den von ihm beigebrachten ERA Stoff nicht zu verstehen! Dank Professor Wille habe ich eine neue Leidenschaft für Prozessoren und VHDL gefunden, es ist immer eine Freude an der ERA Vorlesung von Professor Wille teilzunehmen!

Professor Wille schafft es, mich für ein Thema, das mich wenig interessieren sollte, absolut zu begeistern und komplizierte Inhalte dessen mit sinnvollen Herangehensweisen darzustellen. Klasse!

Der Dozent ist FANTASTISCH. So viel Motivation wünsche ich auch den anderen Dozenten. Er erklärt gut, stellt immer Nachfragen, kreiert eine entspannte Atmosphäre zum Lernen und vermittelt den Studierenden, dass alle es schaffen können, den Inhalt zu verstehen.

Prof. Wille weiß definitiv, wie man die Aufmerksamkeit der Studenten auf sich zieht, den Lernprozess unterhaltsam gestaltet, eine angenehme Umgebung für alle schafft und sehr schwierige und komplexe Dinge Schritt für Schritt klar und verständlich erklärt. Ich freue mich schon sehr auf die Vorlesung. Außerdem sorgt er ständig dafür, dass alle mit dem Stoff, den wir lernen, in Kontakt sind, und er sammelt auf jeden Fall Feedback und gibt die wichtigsten Punkte der Vorlesung wieder.

Sehr motivierend, kann anschaulich erklären und man merkt eine Leidenschaft sowohl für den Lernfortschritt der Studierenden als auch für den behandelten Stoff. Kann gut komplizierte Zusammenhänge herunterbrechen und auch die Folien sind übersichtlich gestaltet und aufs nötigste begrenzt. Mir gefällt auch, dass er schrittweise Überlegungen interaktiv darstellt anstatt nur den Inhalt der Folien vorzutragen.

Ich habe noch nie so viel Spaß bei einer Vorlesung gehabt. Der Dozent ist wirklich in seinem Element und schafft es sehr gut Interesse zu wecken.

Der Dozent kennt sich mit den Inhalten der Vorlesung sehr gut aus, spricht viel frei, versteht, wie die jungen Leute denken und begeistert mich für die Themen aus der Vorlesung.

Es wurde zum Fragen ermutigt, gelobt und auf so gut wie jede Frage wurde eingegangen. Herr Wille konnte mich unfassbar gut für das Fach begeistern und hat sich von allen anderen Dozenten abgehoben. Der Studierende wurde an die Hand genommen und mit Spaß durch die Inhalte der Vorlesung geführt.

Unter Professor Wille wurde ERA zu meiner absoluten Lieblingsvorlesung. Mir war im Vorhinein nicht klar, wie spannend es sein könnte über Addierer und Subtrahierer zu lernen. Ich freue mich jeden Montag und Dienstag aufs neue auf die Vorlesungen, finde die Inhalte sehr spannend und Professor Wille einen großartigen Vortragenden.

Der Dozent zeigt deutliche Begeisterung für das Thema und regt diese auch an. Das "gemeinsame" Erarbeiten der Inhalte trägt sehr um Verständnis bei. Es wird regelmäßig und umfänglich auf Fragen eingegangen und nachgefragt, ob alles verstanden wurde.

Der Dozent ist einfach ein motivierender und lustiger Mensch.
Der Unterhaltungswert der Vorlesungen gleicht eher einem TED Talk. Zudem ist der Stoff sehr gut aufgebaut.

Prof. Dr.-Ing. Robert Wille ist ein sehr energiegeladener Professor und schafft es, den Stoff auf unterhaltsame Weise zu vermitteln. Er hat einen Sinn für Humor, der die Studenten dazu motiviert, den Stoff zu verstehen, während er gleichzeitig sehr engagiert ist und es wirklich so aussieht, als ob er gerne unterrichtet. Er tut dies auch sehr gut, er gibt immer sehr gute Beispiele (praktisch und theoretisch), die helfen, den Stoff zu verstehen. Er lässt die Studierenden immer viele Fragen stellen, um den Stoff zu verdeutlichen.

Ich habe selten einen so guten und begeisterten Dozenten erlebt! So eine Begeisterung auch für die Grundlagen seines Faches zu behalten und das auch so an seine Studenten weiterzugeben ist einfach bemerkenswert! Welcher andere Dozent bekommt bitte spontanen Applaus vom Auditorium, wenn er einen Halbaddierer gebaut hat!?!

Ich habe noch nie so viel Spaß bei einer Vorlesung gehabt. Der Dozent ist wirklich in seinem Element und schafft es sehr gut Interesse zu wecken.

Teaching Awards

In just three years, we have not only established our chair in teaching but also excited our students. This is reflected in outstanding teaching evaluations (see previous page) and the fact that our students have already honored us with three teaching awards.

We are particularly proud to have impressed both our Computer Science and our Electrical Engineering students. A huge and heartfelt "thank you" to all our students for the great vibe in the lecture hall and this fantastic recognition!

Winter Term 2022/23: Lecturer Prize

Already in our first year, we received this award for our lecture "Introduction to Emerging Computing Technologies" from the students of Electrical Engineering.

TEACHING AWARDS

Winter Term 2022/23: TeachInf Award

In the same year, the students of Computer Science awarded us for "Introduction to Computer Architecture" (a lecture with more than 1,000 students!) as the "Best Mandatory Computer Science Lecture."

Winter Term 2023/24: TeachInf Award

One year later, we won the award again, with the students of Computer Science recognizing "Introduction to Computer Architecture" for the second year in a row.

Foto: MQV | Jan Greune

Professional Services 3

Work for Conferences and Journals

Conferences are among the most important platforms for scientists to exchange the latest research findings. In addition to high-quality papers and engaging presentations, a pleasant environment for professional discussions is crucial.

Our Chair is deeply involved in organizing conferences. Furthermore, we are doing service to scientific journals that allow for more in-depth descriptions of scientific findings. This page provides an overview of our work for conferences and journals.

Organization of Conferences and Journals

Design, Automation & Test in Europe
The *Design, Automation and Test in Europe* (DATE) conference is the most important Design Automation Conference in Europe – frequently attracting around 1,000 peers from the community. For several years, we have served on the DATE Executive Committee. In 2023, we served as Program Chair, and since 2024, we have been the Head of the *European Design and Automation Association* (EDAA), the organization that runs DATE.

International Conference on Computer-Aided Design
The *International Conference on Computer-Aided Design* (ICCAD) is the premier forum for Computer-Aided Design. Over the past years, we have served in various roles, including Workshop Chair, Special Session/Tutorial Chair, and (Vice) Program Chair for this esteemed event. Traditionally held in the USA throughout its 44-year history, ICCAD will take place in Munich for the first time in 2025 – with us serving as General Chair.

IEEE Quantum Week
IEEE Quantum Week and its *International Conference on Quantum Computing and Engineering* (QCE) is a relatively new venue experiencing a steep growth. The event bridges the gap between the science of quantum computing and the development of an industry surrounding it. This highly interdisciplinary conference brings together quantum researchers, scientists, engineers, entrepreneurs, developers, students, practitioners, educators, programmers, and newcomers. Since 2024, we are serving on the Organizing Committee of this venue.

Associate Editor for Journals
In addition to our involvement in conferences, we are also actively engaged in the organization of journals. As Associate Editors for leading Design Automation journals such as the *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (TCAD), *Design & Test*, and *Integration, the VLSI Journal*, we oversee the review processes and help define the direction of these important publication outlets.

Service for Program Committees

A crucial component of many scientific conferences is the Program Committee. This committee meticulously reviews and evaluates scientific papers before they are published. Only when the Program Committee is convinced of the contribution and the plausibility of the results is a paper accepted for presentation at a conference. Our chair has been an active member of the most important conferences of our field (e.g., DATE, ICCAD, DAC, ASP-DAC, Quantum Week, and many more) for many years. In the past three years alone, we have served on more than 50 Technical Program Committees.

Selection of Program Committees

Organization of Events for Early-Career Researchers

Conferences provide an excellent platform for early-career researchers to gain deep insights into the state of the art, connect with the community, and build their own network. To this end, most conferences offer dedicated exchange events to facilitate these connections. Our chair has been actively involved in organizing such events. Examples include the *PhD Forum* at the *Design, Automation & Test in Europe* (DATE) conference and the *Design Automation Conference* (DAC) as well as the *ACM Student Research Competition* at the *International Conference on Computer-Aided Design* (ICCAD).

Community Building

Work for existing venues is important, but particularly for emerging technologies, established communities often do not exist or are scattered. Moreover, in research, new topics frequently get introduced and require proper means to get involved. In addition to providing exchange opportunities, this also necessitates a solid foundation, such as funding opportunities. In our work, we are actively involved in several activities that contribute to these needs, helping to build communities and bring together complementary communities with common goals.

Special Sessions, Tutorials, Panels

Our chair is dedicated to fostering community building through organizing special sessions, tutorials, and panels. *Special sessions* allow us to cover "hot topics" and emerging research areas, providing a platform for focused discussions. *Tutorials* offer opportunities to introduce and educate the community on these topics, ensuring that both newcomers and experienced researchers can gain valuable insights. *Panels* facilitate discussions on the directions, impact, and future of these research areas. Over the years, we organized numerous such events, significantly contributing to the formation and growth of communities around new and emerging research topics.

Work for Funding Agencies, Advisory Boards, etc.

In addition to our community-building efforts, we actively engage with funding agencies and advisory boards. Funding plays a crucial role in steering the direction of research communities, and we contribute by reviewing research proposals, advising funding agencies, and discussing strategic directions and programs. Our involvement extends to advisory boards of companies, where we provide insights, feedback, and guidance on how to best consider and utilize recent research results. Through these activities, we help shape the future of research and ensure that emerging technologies receive the support and direction they need, ultimately ensuring that research results find their way into industrial practice.

Munich Quantum Software Forum

Quantum computing is one of the most promising emerging technologies. With corresponding hardware evolving from initial basic research to first actual quantum computing machines, software for this promising technology is becoming key for successful utilization. Numerous players frequently introduce new software solutions, but the community is scattered. To address this, we established the *Munich Quantum Software Forum* (MQSF). The MQSF brings together the "who's who" in quantum computing software for a two-day exchange meeting.

Started in 2023, the MQSF has been a great success. Each event features numerous software presentations through talks and pitches, attracting more than 200 participants. Half the participants come from industry and the other half from academia, with those from the industry further divided equally between established companies (such as IBM, Intel, NVIDIA, BMW, SAP, and more) and start-ups (such as planqc, IQM, Xanadu, AQT, Classiq, and more). Plenty of time is also dedicated to exchanges, hands-on sessions, and discussions.

Due to the huge interest (and the fact that, due to space restrictions, we could not provide everyone with a guaranteed spot), we also recorded all presentations. Together with the presentation slides and a tag cloud created right after the talks by the audience, all these recordings are accessible online (see links below).

MQSF Webpage:
www.cda.cit.tum.de/research/quantum/mqsf

Summary Video of 2024-Edition:
youtu.be/DRbCLtQWTls

Summary Video of 2023-Edition:
youtu.be/x99N7uOKJ1U

MUNICH QUANTUM SOFTWARE FORUM 24-25th of OCTOBER 2024

Service at TUM

We are not only serving the community but also our home university. In addition to the "usual" participation, e.g., in appointment committees, student selection boards, etc., we are particularly involved in the development of our school and the support of doctoral candidates.

Vice Dean for Research and Innovation

Since October 2022, approximately half a year after his appointment as a professor at the Technical University of Munich, Robert Wille was elected Vice Dean for Research and Innovation at the TUM School of Computation, Information and Technology. In this capacity, he oversees and shapes the processes and rules for doctoral candidates to obtain their degrees, helps shape the research direction of the school, fosters international exchange, and much more.

Representative of Doctoral Candidates

As an elected doctoral representative at the TUM School of Computation, Information and Technology, Yannick Stade has been actively serving for three years. With the other representatives, he represents the interests of doctoral students, including mediating between doctoral candidates and their supervisors when issues arise. His role involves ensuring that the voices of his peers are heard and their concerns addressed effectively.

Doctoral candidates (from 2023) of the TUM School of Computation, Information and Technology

Service Award

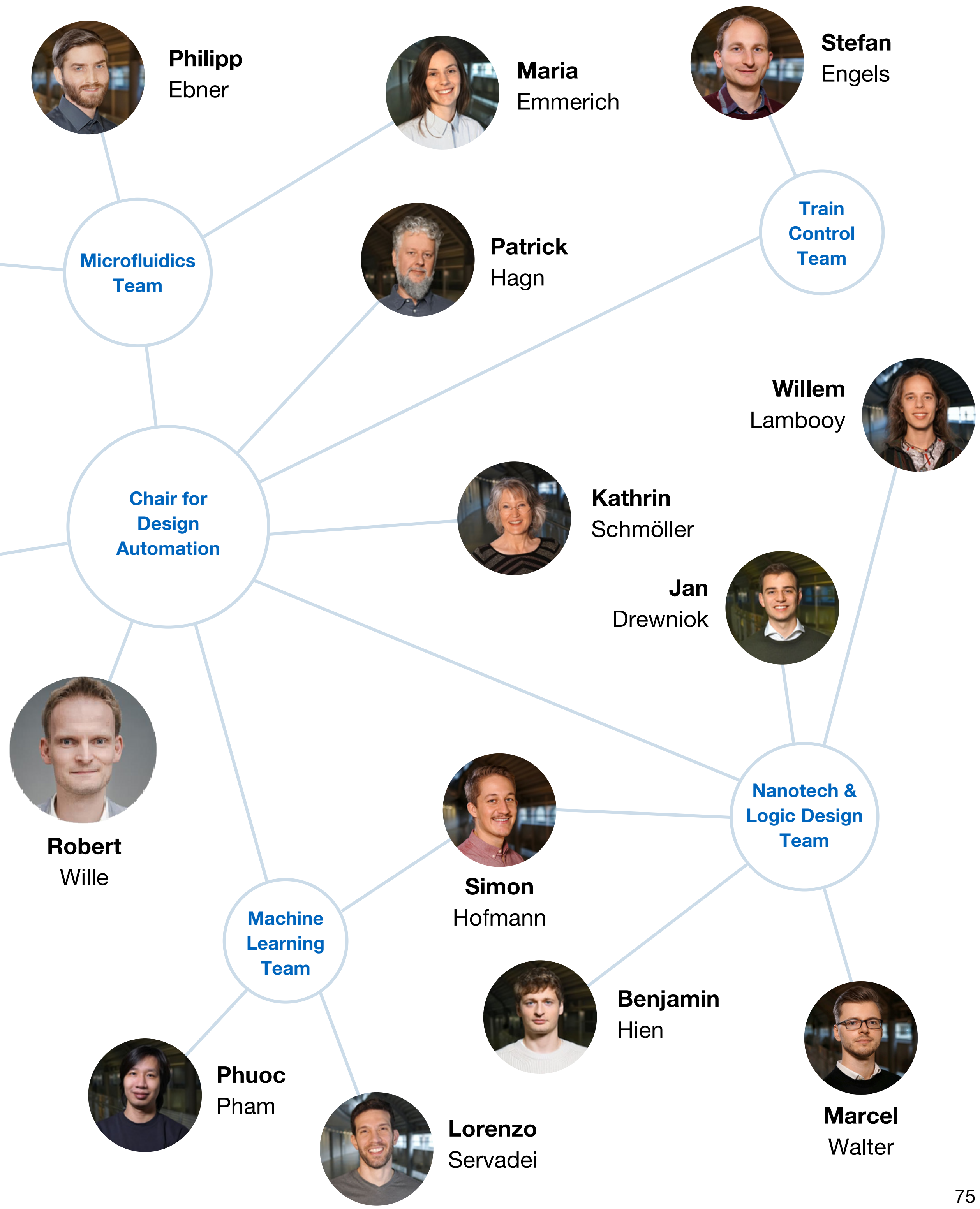
As summarized above, we are very active in serving the community and our university. In July 2023 at the *Design Automation Conference*, some of these activities were recognized with the *ACM SIGDA Meritorious Service Award*. This award was given for "leading positions in major ACM SIGDA conferences." We are deeply honored by this acknowledgment and would like to express our gratitude. We look forward to continuing our service to the community in the future.

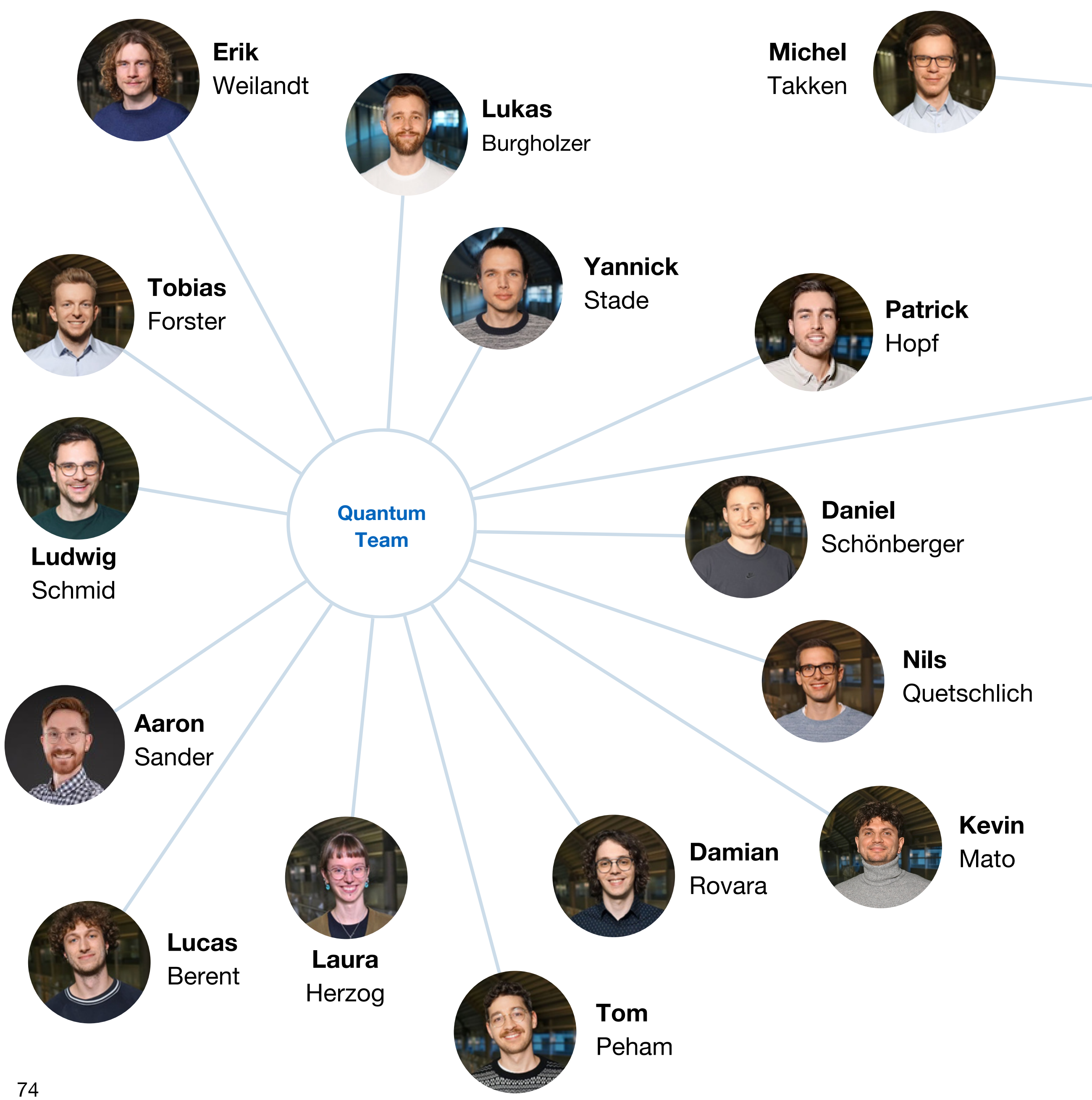
Foto: MQV | Jan Greune

The Team in Action 4

The Team

We are an active and lively group which is passionate about science. We are working in an environment that may best be characterized by the passion to accomplish something new – complemented by teamwork and fostering personal relationships. From assistants, students, researchers, postdocs to professors – we are all working hand in hand and are highly committed to our work. We know how to celebrate our successes, but also how to get through setbacks together! This is the team of the Chair for Design Automation.





The Team at Work

Foto: MQV | Jan Greune

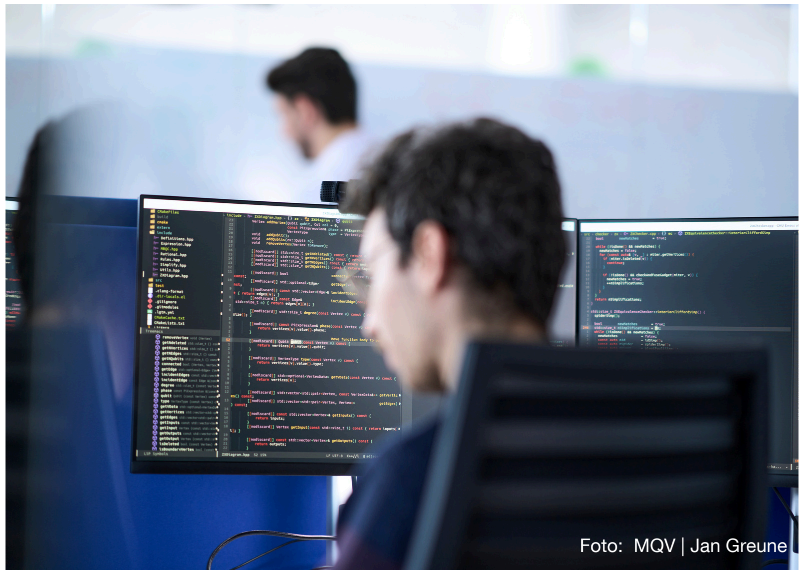


Foto: MQV | Jan Greune

Foto: MQV | Jan Greune

This is "our kingdom" – the place where creativity thrives and design automation solutions for today's and emerging technologies are born. Since our primary goal is to develop innovative methods and software, much of our work involves sitting in front of computers. However, we also have spaces for discussions, exchanges, and creative brainstorming – and even a lab to manufacture small prototypes. Our offices, including our "Quantum Space" and "Munich Microfluidic Lab", provide the ideal environment for all these activities.

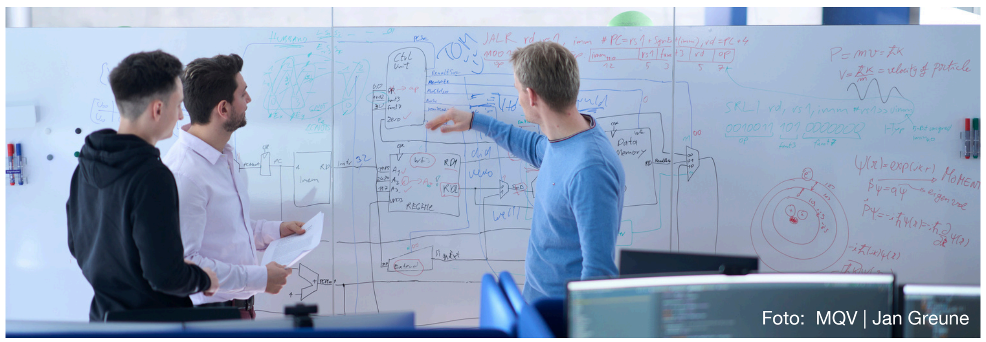


Foto: MQV | Jan Greune

The Team at Conferences

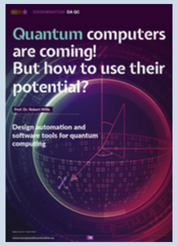
Conferences are where we present our work, connect with the community, get inspiration for new endeavors, and foster as well as create new collaborations. These exchanges are key to our successful work and, hence, it is essential that we are represented here. This page provides some impressions of us being at conferences.

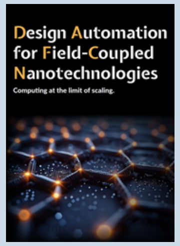
The Team Reaching Out

It is important to us that our work does not remain confined to the "ivory tower". We aim to share our results with others and further develop them collaboratively. In particular, we are committed to making our work, which is largely funded by public resources, understandable to the general public. We achieve this through popular science articles, videos, podcasts, exhibitions, school visits, and more. We believe that even complex topics can be presented in a way that everyone understands. This page summarizes some of our outreach activities.

Popular Science Articles

While our primary audience is the scientific community, with over 150 scientific articles published in peer-reviewed journals and conference proceedings in the past three years, we also strive to reach the general public. To achieve this, we have created popular science articles that explain our work in layman's terms. Examples of these articles, displayed on the right-hand side, cover topics such as quantum computing, microfluidics (particularly labs-on-a-chip), and nanotechnologies. These articles aim to make complex scientific concepts accessible and engaging for everyone.

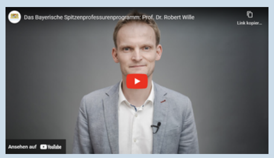




Videos, Podcasts, etc.

A picture is worth a thousand words. And a video, perhaps even more. Accordingly, we also publish videos of our work from time to time, often in collaboration with our partners. These videos briefly explain our mission, showcase current research topics, summarize events and talks, and provide insights into our research and teaching landscape. In addition to videos, we also are in podcasts occasionally – allowing us to discuss various aspects of our work and offering another engaging way to reach out.

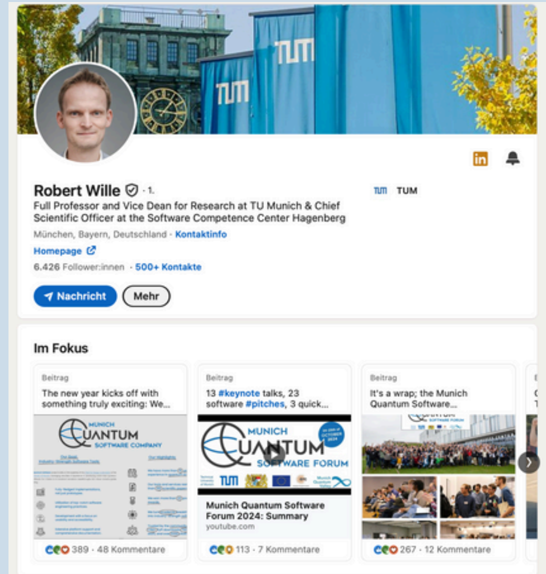
Interested in having a look? Check out this page: www.cda.cit.tum.de/videos/



LinkedIn Profiles

Social media is another great channel to reach out and present our work – especially professional networking platforms such as LinkedIn. Accordingly, we are active here as well. Interested in connecting? Reach out to us:

Robert Wille: robertwille
Marcel Walter: dr-marcel-walter
Lukas Burgholzer: lukas-burgholzer-7a1741a7
Lorenzo Servadei: dr-lorenzo-servadei
Kevin Mato: kevin-mato-657444173
Nils Quetschlich: nilsquetschlich
Michel Tacken: michel-takken-b2025a167
Philipp Ebner: philipp-ebner-3a1376344
Daniel Schönberger: daniel-schoenberger-18326b1b7
Jan Drewniok: jan-drewniok-6534bb218
Maria Emmerich: maria-emmerich-1b2b981a8
Stefan Engels: s-engels
Simon Hofmann: simon-hofmann-
Aaron Sander: aaron-sander
Damian Rovara: damian-rovara
Phuoc Pham: phuocphn
Yannick Stade: yannick-stade-33787b21b
Tobias Forster: tobias-forster-496a0626b
Erik Weilandt: erik-weilandt
Laura Herzog: laura-s-herzog-122b68252
Willem Lambooy: willem-lambooy-056b6a164
Patrick Hopf: hopf-patrick



Expos and Exhibitions

Direct interactions are invaluable! While we regularly engage with the scientific community at conferences, reaching broader audiences is often not possible in those settings. Therefore, whenever possible, we also try to participate in and present at expos and exhibitions. These events allow us to connect directly with a broader audience, enabling us to extend our reach even further.

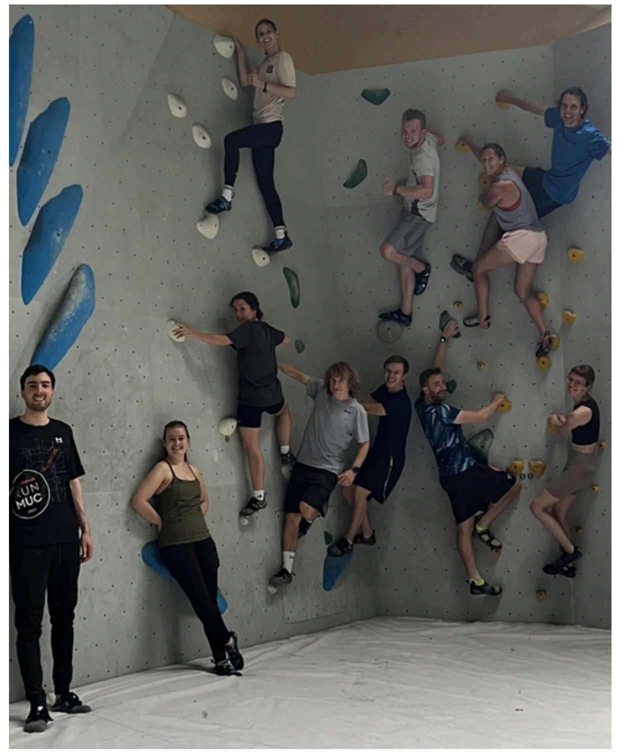
Further Activities

Besides that, there are many further outreach activities we are engaging in – often together with our partners. Examples include the MQV In Persona series, the Quick Quantum Questions, the Hack4Her series, or series and initiatives from projects and partners.

Interested in doing some outreach together? Reach out to us!

The Team Off Duty

We work hard, and we play hard. Whether it is jogging, skiing, hiking, ice skating, bouldering, bowling, foosball, cooking, planting, or any other activity, we are on board! The same goes for attending typical Munich festivities such as Oktoberfest and Frühlingsfest – embracing the Bavarian stereotype and connecting cultures worldwide. This page provides some impressions of our off-duty activities.



Impressum

Publisher:	Chair for Design Automation (Prof. Dr. Robert Wille) Arcisstr. 21, 80333 München, Germany Tel.: +49 (0) 89 289 23551 E-Mail: robert.wille@tum.de
Editors:	Robert Wille and Team
Design:	Simone Hauser, Patrick Hagn, Yannick Stade
Print:	printy A.Wittek GmbH, Luisenstr. 49, 80333 München
Publication:	2025