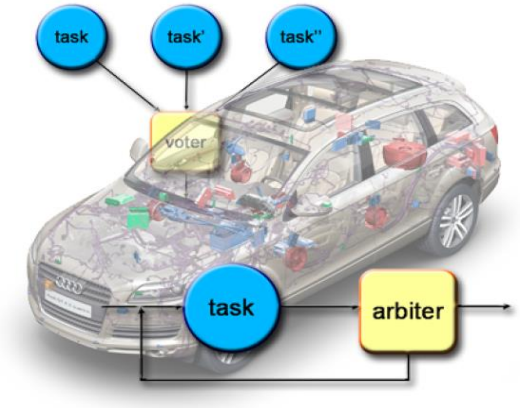
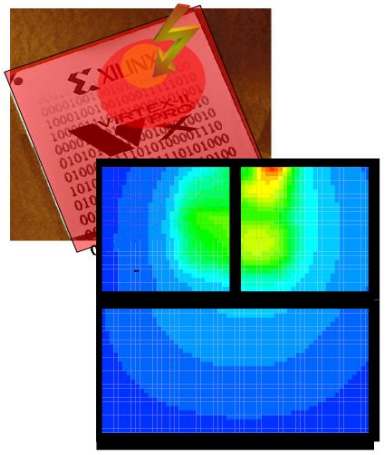
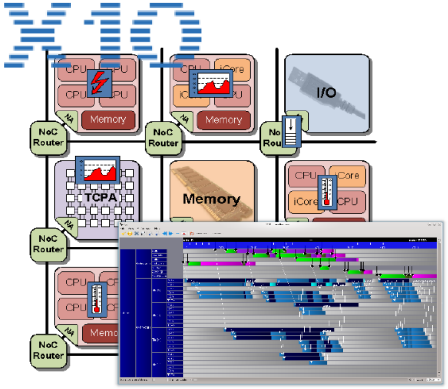
Topics for Project, Bachelor's and Master's Theses



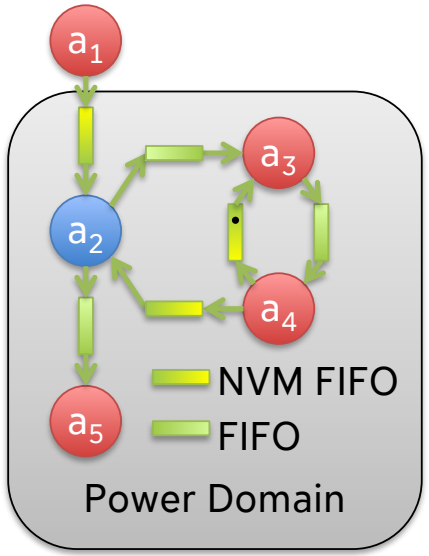


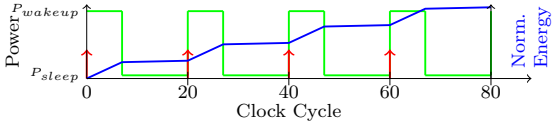


https://www.cs12.tf.fau.eu/teaching

Performance Optimization of Self-powering DFGs

Dataflow networks are ideally suited to model stream processing applications, and can be used in IoT devices signal for processing and compression. Recent research on *Self-powering Dataflow Networks* have shown that power savings can be achieved on such networks by powering down actors during periods of data unavailability using clock-gating and power-gating techniques. This is done by systematically transforming the FSM of each actor and introducing sleep states. However, this may lead to adverse effects on the throughput and even the total energy consumption of the network due to the additional latency incurred for sleep and wake-up transitions, as well as additonal power consumption due to the overhead circuitry introduced by the transform.





The goal of this thesis is to explore sleep and wake-up strategies, to optimize for throughput and energy consumption. Such strategies may include introducing optimal delays before going to sleep based on the dataflow specifications, and/or creating power domains containing multiple actors and channels to reduce additional circuitry overheads.

Prerequisites:	C++ and VHDL or Verilog knowledge required
Type of Work:	Theory (30%), Conception (30%), Implementation (40%)
Supervisor:	Abrarul Karim (abrarul.karim@fau.de) and Joachim Falk (joachim.falk@fau.de)

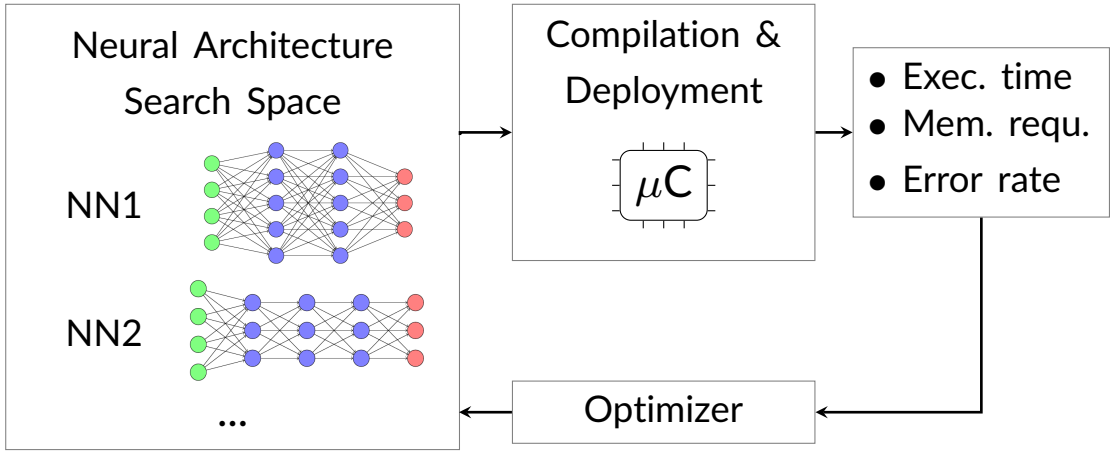
design

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

FAU
Friedrich-Alexander-Universität
Technische Fakultät

Neural Architecture Search for Time Series Prediction on µCs

Deploying neural networks (NNs) on microcontrollers (µCs) allows AI applications to be run close to sensors and increases the scope for future applications. However, designing NNs for resource-constrained µCs requires a delicate balance between maintaining a low prediction error while achieving low memory consumption and execution time.



As this is extremely challenging to achieve manually, platform-aware Neural Architecture Search (NAS) has become a major research topic. Here, the NNs are optimized regarding, e.g., the number of layers or the number of neurons per layer, with the aim of reducing the error rate and the execution time of NNs for a given target platform. In this work, one NAS algorithm is applied to the problem of time series prediction for an automotive µC as target platform. Finally, the NNs found with NAS are compared with existing neural networks and evaluated in terms of NN error rate, as well as memory consumption and execution time when deployed on the target µC.

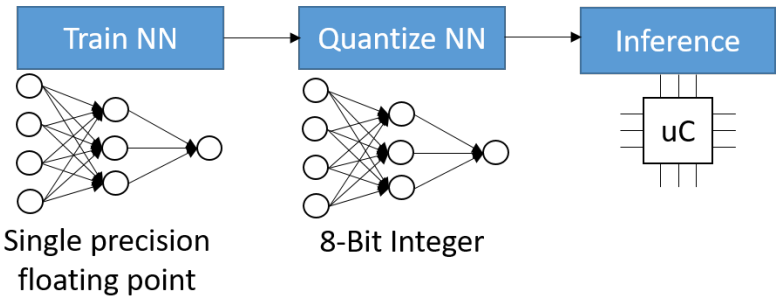
Requirements:	Knowledge in C, Python, and Neural Networks
Type of thesis:	Theory (20%), concept (40%), implementation (40%)
Supervisor:	Christian Heidorn (Christian.Heidorn@fau.de)

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

Friedrich-Alexander-Universität
Technische Fakultät

Exploring Quantization of DNNs for Regressions Tasks

Quantization of *Neural Networks (NNs)* is a common way to reduce their computational intensity, which is particularly useful when using resource-constrained microcontrollers (µC). While quantization of NNs trained on classification tasks has been documented to work well, quantization of NNs trained on regression tasks often results in significant degradation of the NN's prediction performance (accuracy).



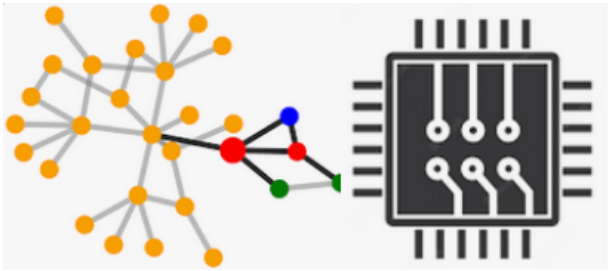
The main reason for this observation is that for classification tasks, small changes in the predicted probability vector across all classes due to quantization usually do not affect the predicted class derived from the distribution. In contrast, for regression tasks, small changes in the prediction can lead to a significant decrease in accuracy. The goal of this thesis is to investigate, implement and compare DNN training and quantization methods that can help improve the accuracy of quantized DNNs in regression tasks. Furthermore, the selected techniques should be evaluated in terms of the accuracy achieved for different regression problems as well as required memory consumption and execution time when deployed on different µCs.

Requirements:	Knowledge in C, Python, and Neural Networks
Type of thesis:	Theory (20%), concept (40%), implementation (40%)
Supervisor:	Christian Heidorn (Christian.Heidorn@fau.de), Mark Deutel (mark.deutel@fau.de)

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

Friedrich-Alexander-Universität
Technische Fakultät

Towards ML-Guided Integrated Circuit Synthesis



An important step in Electronic Design Automation (EDA) is the synthesis of the Register Transfer Level (RTL). In this step, a Hardware Description Language (HDL) code is converted into a netlist using various optimizations. These optimizations use various heuristics and minimize the size, power, and speed of the final Integrated Circuit (IC). Graph Neural Networks (GNNs) have been successfully applied to combinatorial optimization problems. Recently, the OpenABC dataset was released to spur research in this area.

The dataset consists of a large set of netlists using various scenarios. As synthesis runs are computationally costly, Machine Learning (ML) can be used to predict the quality of the synthesis recipe without carrying out the synthesis. A set of simple graph models have been benchmarked on the OpenABC dataset.
The following tasks shall be carried out as part of this thesis:

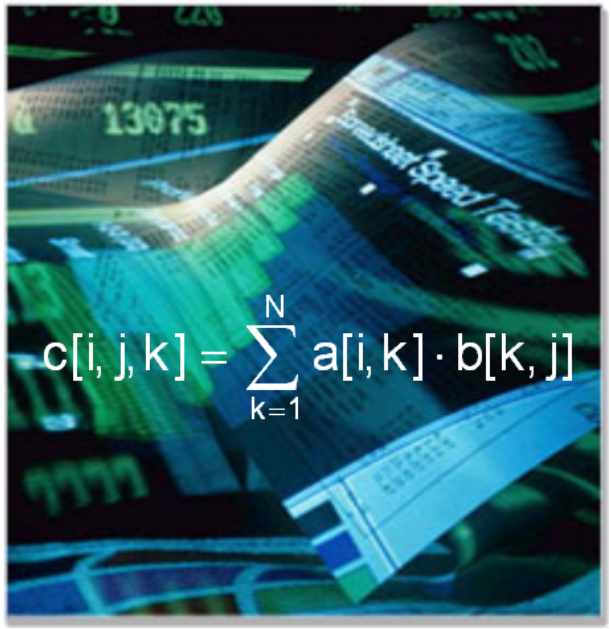
1. Reproduce the existing results.
2. Develop novel models to improve the prediction quality.
3. Propose additional benefits of applying ML for tasks related to IC synthesis.

Prerequisite:	Programing Skills in Python, Understanding of VHDL and Circuit Design
Type of Work:	Theory (20%), Concept (30%), Implementation (50%)
Contact Person:	Muhammad Sabih (muhammad.sabih@fau.de)

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

Friedrich-Alexander-Universität
Technische Fakultät

Data Integrity Modeling and Assurance in Co-Design



The assurance of security requirements, i.e., the integrity of data exchanged between resources in an embedded system, has not yet been adequately considered in design automation.
Data integrity can be impaired by either faults, aging, but also by security attacks. Examples of countermeasures include techniques for fault detection and fault correction, e.g., error-correcting codes, ECCs, redundancy, or through strong isolation of application data from other applications, e.g., by memory access control.
In this Master thesis, constraints on the integrity of data should be incorporated into an established co-design flow by attributing communications between tasks with integrity attributes. Based on these attributes, mapping constraints shall be generated to establish and thus assure integrity constraints to hold in any feasible mapping, including the allocation of resources and binding of tasks to resources.
The constraints to be generated thereby restrict mapping options, e.g., that data items of different applications must not be mapped to the same memory (strong isolation) or that memory protection is enabled through memory mapping units (MMUs). The techniques shall be implemented in an existing co-design framework and tested for sample applications.

Prerequisites:	Java knowledge required
Type of Work:	Theory (40%), Conception (20%), Implementation (40%)
Supervisor:	Joachim Falk (joachim.falk@fau.de) and Jürgen Teich (juergen.teich@fau.de)

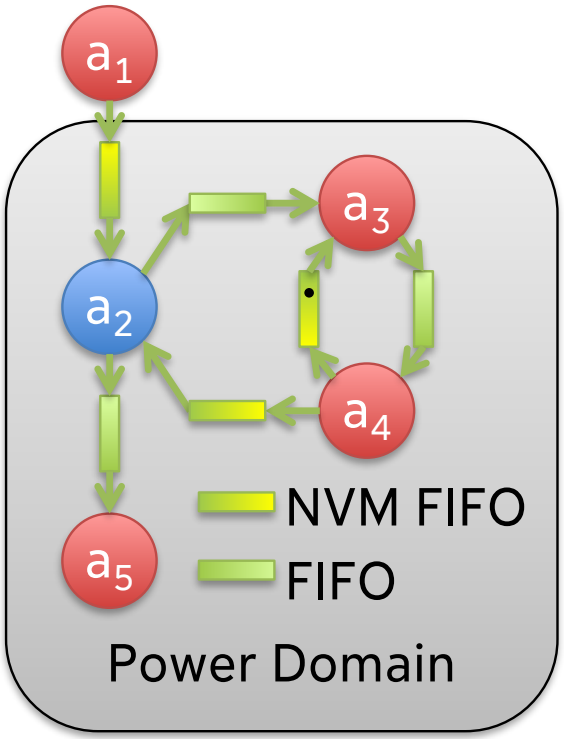
codesign

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

FAU
Friedrich-Alexander-Universität
Technische Fakultät

Exploiting Non-Volatile Memory for Dataflow Computation

IoT sensor platforms need to be cheap, (ultra-)low power, long-running, and maintenance-free. Thus, battery-less IoT devices utilizing energy harvesting are more and more deployed. Often, such systems execute signal processing applications, which can be specified preferably by dataflow networks. These naturally allow the exploitation of concurrency by implementing each actor as a hardware circuit, all running in parallel. However, programming such sensor platforms offers unique challenges due to their intermittent power supply.



In this thesis, Non-Volatile Memory (NVM) should be exploited to realize dataflow networks in hardware to tackle these intermittency issues, e.g., by investigating and modeling persistable FIFO-based memory units. In particular, dataflow networks operating in mixed volatile/non-volatile operating modes shall be modeled by combining the system-level concept of dataflow, which is based on self-scheduled activations of computations, with NVM-based FIFOs. Inactive actors or even subnets should power down and reactivate upon the arrival of more data to be processed. In addition, for a continuously safe mode of operation, a powering down must also be triggered upon any intermittent shortage of power supply. Analogously, actors shall perform an auto-wakeup after recovery from the power shortage.

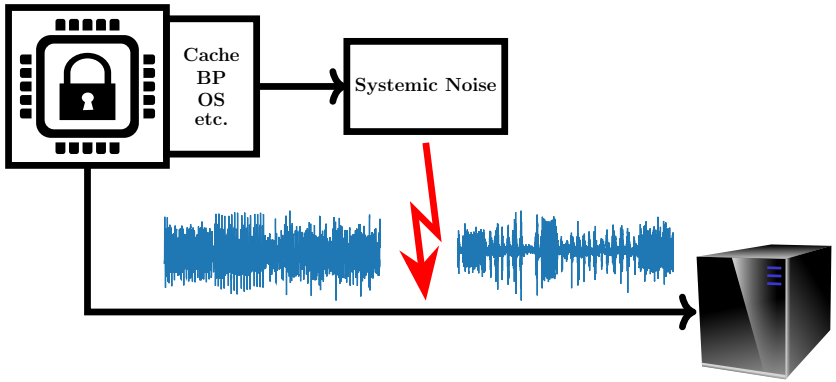
Prerequisites:	C++ and VHDL or Verilog knowledge required
Type of Work:	Theory (30%), Conception (30%), Implementation (40%)
Supervisor:	Joachim Falk (joachim.falk@fau.de)

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

Friedrich-Alexander-Universität
Technische Fakultät

Investigating the Effects of Systemic Noise in physical Side-Channel Analysis on High-Performance Targets

Side-channel attacks are a powerful tool to extract secret information from cryptographic algorithms otherwise considered secure. However, side-channel analysis research is often performed on rather simple hardware, like microcontrollers, making the resulting attack techniques not necessarily applicable on more complex platforms.



This work focuses on the *systemic noise* inherent to complex platforms, as they contain performance-enhancing features, like Caches or Branch Prediction, which may lead to less deterministic execution behavior and as such less consistent side-channel measurements, impeding attacks.

In this work, a physical side-channel analysis of current cryptographic algorithm standards implemented on a high-performance platform containing multiple sources of *systemic noise* is performed in order to research the impact of the different *systemic noise* components and the applicability of previously developed side-channel attacks. Furthermore the extraction of internal data from associated *systemic noise* and its usage as another exploitable side-channel for attacks is investigated.

Prerequisites:	Basic knowledge in C/C++ and Python
Type of Work:	Theory (30%), Conception (20%), Implementation (50%)
Supervisor:	Paul Krüger (paul.krueger@fau.de)

design

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

FAU
Friedrich-Alexander-Universität
Technische Fakultät

Runtime Requirement Enforcement of Safety Properties of Human-Computer Interaction

Ensuring human safety is the most important factor within the area of human-robot interaction (HRI). E.g., robotic assistants should safely and flexibly cooperate with human operators. In addition, it must be guaranteed that they do not collide with humans and any obstacles in their shared environment.
Runtime Requirement Enforcement (RRE) has been proposed to ensure the satisfaction of system properties in the presence of uncertainty represented by the varying input from the environment. Finite state machines (FSMs) are used to model the enforcement strategy, which allows to perform formal verification and consequently provide safety guarantees.
In this thesis, an FSM-based RRE should be developed for enforcing a given set of safety requirements on simple HRI use cases such as a handover task. Formal verification (e.g., via PRISM verification tool) should be conducted to provide formal guarantees for the developed enforcement strategy to satisfy the given safety requirements.

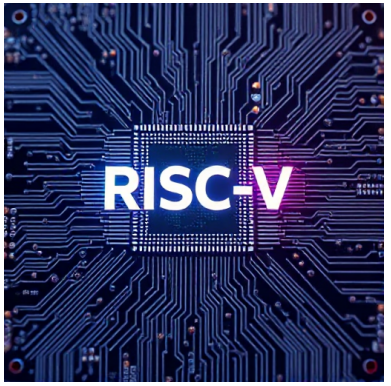
Voraussetzungen:	Programming knowledge in C++ or Java, knowledge in formal verification and temporal logic, basic knowledge in Robotics, basic knowledge in control
Art der Arbeit:	Theory (30%), Conception (35%), Implementation (35%)
Ansprechpartner:	Khalil Esper (khalil.esper@fau.de)

design

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

FAU
Friedrich-Alexander-Universität
Technische Fakultät

Efficient Hardware/Software Co-Design for Deploying Neural Networks on FPGAs



Different neural networks and machine learning algorithms have varying complexity, characterized by number of computations, memory requirements for storing model and intermediate data. To optimize efficiency, hardware configurations may be tailored to such specific workloads. The growing interest in open-source hardware and toolchains has led to the development of customizable RISC-V cores, which can be implemented on FPGAs as "soft-cores". These cores offer flexible configuration features like cache sizes, bus widths, and hardware extensions. However, the complex design space makes optimization challenging and requires a careful hardware/software co-design of neural network models and hardware architectures.

The primary goal is to develop a systematic design exploration strategy that effectively navigates the trade-offs between area, latency, and performance. The methodology involves deploying and profiling neural network workloads on soft-core RISC-V implementations to establish baseline performance metrics. Subsequently, the research will explore dedicated hardware accelerators and various hardware configurations, aiming to optimize the balance between resource utilization and computational efficiency.

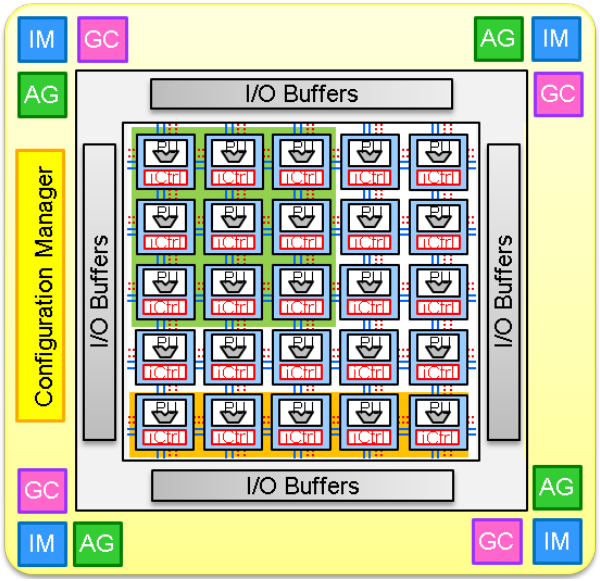
Prerequisites:	Knowledge in Python, C/C++, and hardware design
Type of Work:	Theory (10%), Conception (30%), Implementation (60%)
Supervisors:	Batuhan Sesli (batuhan.sesli@fau.de)

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

Friedrich-Alexander-Universität
Technische Fakultät

Automatische Exploration von Varianten eng gekoppelter Prozessorfelder

Tightly Coupled Processor Arrays (TCPAs) bestehen aus einem Feld von leichtgewichtigen Rechenelementen (PEs), die über ein rekonfigurierbares Netzwerk eng miteinander gekoppelt sind. TCPAs stellen eine ganze Klasse unterschiedlicher Beschleunigerarchitekturen dar, die über eine Menge von Parametern (Anzahl PEs, Speichergröße, usw.) definiert ist. Dabei hängen wichtige Architektureigenschaften wie die elektrische Leistungsaufnahme, Chipfläche und Rechenleistung oft direkt von den gewählten Parametern ab. Die manuelle Auswahl geeigneter Parameter ist jedoch sehr aufwendig und erfordert ein hohes Maß an Expertenwissen.



Ziel dieser Arbeit ist es daher, diesen Entwurfsraum so zu explorieren, dass für eine gegebene Anwendungsmenge eine maßgeschneiderte Zielarchitektur automatisch generiert werden kann.

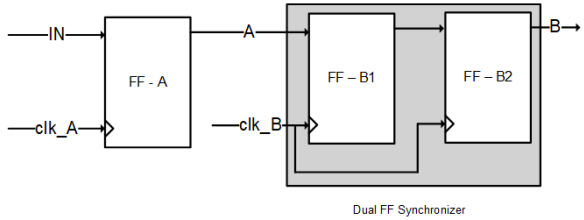
Voraussetzungen:	Sehr gute Kenntnisse in C++- und Python-Programmierung
Art der Arbeit:	Theorie (10%), Konzeption (30%), Implementierung (60%)
Ansprechpartner:	Dominik Walter (dominik.l.walter@fau.de)
	Frank Hannig (frank.hannig@fau.de)

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

Friedrich-Alexander-Universität
Technische Fakultät

Analyse von taktdomänenübergreifender PE-Kommunikation in eng gekoppelten Prozessorfeldern

Tightly Coupled Processor Arrays (TCPAs) bestehen aus einem Feld von leichtgewichtigen Rechenelementen (PEs), die über ein rekonfigurierbares Netzwerk eng miteinander gekoppelt sind. Diese direkte Kommunikation mit benachbarten PEs erfordert allerdings eine synchrone Ausführung aller PEs, weswegen in den meisten Implementierungen das gesamte TCPA taktsynchron realisiert wird. Um eine hohe Taktfrequenz zu erreichen, werden allerdings größere Chips typischerweise in mehrere Taktdomänen unterteilt, die zwar die gleiche Taktquelle, jedoch einen unterschiedlichen Versatz (engl. *skew*) aufweisen. Signale können dabei nicht direkt zwischen unterschiedlichen Taktdomänen ausgetauscht werden, sondern müssen über spezifische sogenannte *Clock Domain Crossings* (CDCs) explizit synchronisiert werden.



https://anysilicon.com/clock-domän-crossing-cdc/

Ziel dieser Arbeit ist es, eine Bewertung von PE-lokalen Taktdomänen für TCPAs zu erstellen. Hierfür müssen in der Arbeit neben der Erweiterung bestehender VHDL-Komponenten auch *Design Constraints* in gängigen FPGA-Entwicklungsumgebungen berücksichtigt werden.

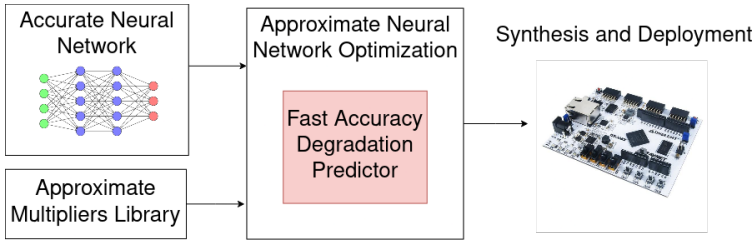
Voraussetzungen:	Sehr gute Kenntnisse in VHDL und FPGA-Programmierung
Art der Arbeit:	Theorie (10%), Konzeption (30%), Implementierung (60%)
Ansprechpartner:	Dominik Walter (dominik.l.walter@fau.de)

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

Friedrich-Alexander-Universität
Technische Fakultät

RISC-V Extensions for Approximate DNNs

Embedded Machine Learning (ML) is a fast-growing field incorporating ML algorithms, hardware, and software to be deployed in resource constrained devices. Approximate computing aims to tradeoff hardware and/or energy resources for inaccurate computations.



This work aims to extend the RISC-V processor with approximate computing units. At first, RISC-V extensions (as tightly coupled units or co-processors) shall be developed for quantized DNNs. Then, approximate units shall be included in the RISC-V extensions. In parallel, the work will explore the possibility of simulating the accuracy degradation introduced by approximate multiplications for various deep-learning models. An existing framework such as Brevitas or AdapT may be used for this purpose. Subsequently, it is also possible to predict the accuracy degradation introduced by the approximate units by using the information of a DNN architecture and the type of approximate unit/s. This prediction of accuracy degradation is expected to be faster as compared to the exact calculation on a test dataset, which will aid in efficient exploration of the design space exploration of approximate functional units.

Prerequisites:	Knowledge in Python, HDL (e.g. Verilog or VHDL) and machine learning
Type of Work:	Theory (30%), Conception (20%), Implementation (50%)
Supervisor:	José Juan Hernández Morales (*jose.juan.hernandez*@fau.de), Muhammad Sabih (*muhammad.sabih*@fau.de)

Lehrstuhl für Informatik 12
Hardware-Software-Co-Design
Cauerstraße 11
91058 Erlangen

Friedrich-Alexander-Universität
Technische Fakultät