Automating FPGA Accelerator Design using Large Language Models

Vinamra Sharma, Xingjian Fu, Jude Haris, José Cano
School of Computing Science, University of Glasgow, Scotland, UK

Abstract

The growing complexity of FPGA accelerator design presents a major challenge for researchers and developers, requiring extensive exploration of architectural trade-offs involving computation, memory organization, and resource utilization in order to design efficient accelerators. Although recent advances in Large Language Models (LLMs) have demonstrated strong capabilities in reasoning and code generation, their potential for guiding hardware design workflows remains underexplored. This short paper presents an emerging research direction towards self-improving FPGA accelerator design through the integration of LLM-guided reasoning and iterative hardware feedback. Building upon the SECDA ecosystem, we explore how retrieval-augmented reasoning, structured design space exploration, and hardware evaluation data can be combined to support automated accelerator design generation and refinement. Preliminary results demonstrate the successful generation and execution of three accelerator designs, providing early evidence that hardware feedback can be leveraged to improve future design decisions. This work highlights the potential of AI-assisted hardware co-design and outlines a path towards more autonomous FPGA accelerator development workflows.

Keywords: FPGA Accelerators; Hardware Design Automation; Design Space Exploration; Large Language Models.

1 Introduction

The increasing demand for artificial intelligence (AI) applications has driven the need for specialized hardware accelerators capable of delivering high performance and energy-efficient computation. Field-programmable gate arrays (FPGAs) are widely used for this purpose due to their flexibility, reconfiguration ability, and capability to support workload-specific optimizations. However, developing FPGA accelerators remains a challenging task that requires navigating through a complex design space, exploring trade-offs between compute parallelism, memory organization, dataflow strategies, and resource utilization. As a result, accelerator development often relies on extensive design space exploration (DSE) that requires iterative computations and significant hardware expertise.

Recent advances in Large Language Models (LLMs) have demonstrated strong capabilities in reasoning, code generation, and engineering assistance across a wide range of domains. These capabilities create new opportunities for applying AI not only to software development but also to hardware design workflows. In particular, LLMs have the potential to assist accelerator development by reasoning about architectural trade-offs, generating hardware designs, and learning from performance outcomes of previous designs. Despite growing interest in AI-assisted engineering, the use of LLMs for FPGA accelerator design and hardware exploration remains relatively limited.

This work investigates the research direction towards self-improving FPGA accelerator design through the integration of LLM-guided reasoning and hardware feedback. Building upon the SECDA ecosystem, this work explores how Retrieval-Augmented Generation (RAG), structured exploration, and evaluation data collected from simulation and FPGA execution can be combined to support automated accelerator generation and refinement over time through fine-tuning based on the execution performance results. Rather than treating hardware design as a purely manual optimization process, this approach enables the continuous accumulation and reuse of knowledge gained through fine-tuning to guide future design decisions.

Initial results obtained through the SECDA-DSE framework demonstrate the successful generation and Xilinx Zynq-7000 FPGA execution of three accelerator workloads, including vector multiplication, 2D convolution, and matrix transpose kernels. These early findings provide evidence that LLM-guided design workflows can support FPGA accelerator development and motivate future research towards more autonomous, adaptive, and AI-assisted hardware design space exploration methodologies.

2 Related Work

SECDA introduced a hardware-software co-design methodology for FPGA-based deep neural network (DNN) accelerators, enabling rapid accelerator development through SystemC simulation, hardware synthesis, and FPGA execution workflows [1]. Building upon this foundation, SECDA-TFLite extended the ecosystem with reusable architectural templates and TensorFlow Lite integration, simplifying the development and deployment of custom FPGA accelerators for edge AI applications [2]. Despite these advances, identifying efficient accelerator configurations and navigating the associated design space remains a challenging and largely manual process that requires multiple iterations.

Recent work has explored the use of Large Language Models (LLMs) to support FPGA accelerator design within the

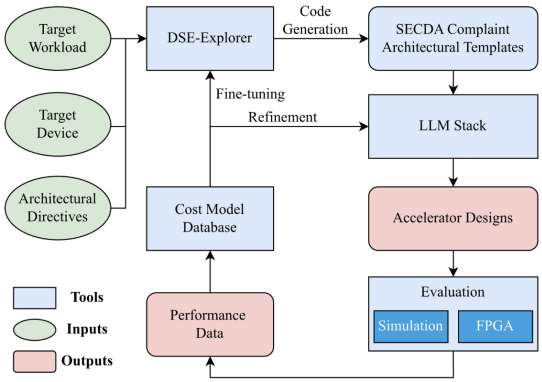


Figure 1. Overview of SECDA-DSE.

SECDA ecosystem. SECDA-DSE introduced an LLM-guided framework that combines retrieval-augmented reasoning, structured design space exploration, and iterative hardware feedback to assist accelerator generation and refinement [3]. Subsequently, FPGA execution of multiple generated accelerator workloads demonstrated the feasibility of using LLM-guided workflows to produce SECDA-compliant accelerator designs and validate them on hardware [4]. Building on these efforts, this work investigates the broader vision of self-improving FPGA accelerator design, where hardware feedback and accumulated design knowledge are continuously leveraged to guide future accelerator development.

3 SECDA-DSE

SECDA-DSE is an LLM-guided framework for FPGA accelerator design space exploration built on the SECDA ecosystem. As shown in Figure 1, the framework combines structured design space exploration, RAG, and hardware evaluation to generate SECDA-compliant accelerator designs. Generated accelerators are then validated through simulation, synthesis, and real FPGA execution, while performance metrics are stored and reused to guide future design iterations. By incorporating hardware feedback into the exploration process, SECDA-DSE provides a foundation for more adaptive and self-improving FPGA accelerator design workflows.

4 Results and Evaluation

To provide preliminary evidence for the proposed approach, SECDA-DSE was used to generate three FPGA accelerator workloads: element-wise vector multiplication, 2D convolution, and matrix transpose. All generated designs successfully passed synthesis and FPGA execution on an Xilinx Zynq-7000 platform. Table 1 summarizes the execution latency and FPGA resource utilization of the generated accelerators. The hardware counter (HWC) cycles report load, compute and store counter values collected during FPGA execution.

The generated accelerators exhibit workload-specific execution and resource utilization characteristics. The vector multiplication accelerator achieved the lowest execution latency (135 ms) while utilizing the highest proportion of DSP resources (21.82%), reflecting its compute-intensive nature. In contrast, the transpose accelerator exhibited higher latency (238 ms) and LUT utilization (8.85%), indicating a stronger dependence on memory movement and data reorganization. The HWC measurements further reveal distinct execution behaviours across workloads, with convolution showing increased buffering and data reuse overheads compared to vector multiplication and transpose. These results provide early evidence that LLM-guided accelerator generation can adapt architectural decisions to different workload characteristics while producing FPGA-executable designs.

Table 1. FPGA execution results of generated accelerators.

Metric	2D Conv	VMUL	Transpose
Validation	Pass	Pass	Pass
Latency (ms)	163	135	238
HWC Cycles	1251/76/1250	52/26/51	393/311/387
BRAM (%)	2.50	3.57	2.86
DSP (%)	1.36	21.82	5.45
LUT (%)	6.64	8.23	8.85

5 Conclusion

This short paper explored an emerging research direction towards self-improving FPGA accelerator design through the integration of LLMs, structured design space exploration, and iterative hardware feedback. Building on the SECDA ecosystem, SECDA-DSE demonstrates how retrieval-augmented reasoning and accumulated hardware knowledge can be leveraged to support automated accelerator generation and refinement. Initial results show the successful synthesis and FPGA execution of the generated accelerator workloads, providing early evidence for the SECDA-DSE feasibility.

Acknowledgments

All authors gratefully acknowledge UK Research and Innovation (UKRI) and Engineering and Physical Sciences Research Council (EPSRC) funding for the AI Hub for Productive Research and Innovation in Electronics (APRIL) AI Hub [grant number EP/Y029763/1].

References

[1] J. Haris, P. Gibson, J. Cano, N. Bohm Agostini, and D. Kaeli. 2021. SECDA: Efficient Hardware/Software Co-Design of FPGA-based DNN Accelerators for Edge Inference. In *SBAC-PAD*.

[2] J. Haris, P. Gibson, J. Cano, N. Bohm Agostini, and D. Kaeli. 2023. SECDA-TFLite: A toolkit for efficient development of FPGA-based DNN accelerators for edge inference. *J. Parallel and Distrib. Comput.* (2023).

[3] V. Sharma, X. Fu, J. Haris, and J. Cano. 2026. LLM-Driven Design Space Exploration of FPGA-based Accelerators. In *InSyDe workshop @ EuroSys*.

[4] V. Sharma, X. Fu, J. Haris, and J. Cano. 2026. Towards Autonomous Accelerator Design: FPGA Accelerator Generation with SECDA. In *MLArchSys workshop @ ISCA*.