

Hackathon  
→ Custom Acceleration of Large Language Model Primitives

Studio Version 9.4.2

Release Date March 2025  
  
Author: Tadej Murovič, [tadej.murovic@codasip.com](mailto:tadej.murovic@codasip.com)

## 

# **INTRODUCTION**

As **large language models (LLMs)** continue to transform cutting-edge **natural language processing** and next-generation AI, the need for faster, more **efficient** computing platforms grows in parallel. At **Codasip**, we recognize that **hardware–software co-design** is the key to unlocking the performance gains needed to keep pace with the ever-increasing demands of AI.

This hackathon will leverage **Codasip Studio** in the cloud to illustrate how **LLM backends** (in particular, the **Transformer encoder’s linear algebra operations**) can be optimized through **software–hardware codesign**. You’ll explore:

1. **Software Optimization**: Rewrite the Transformer encoder code to handle its linear algebra tasks more efficiently.
2. **Hardware Enhancement**: Design custom **RISC-V CPU instructions** specifically targeted for these operations.
3. **Co-Design Integration**: Invoke the new hardware instructions within your revised software, creating a seamless interplay between the two layers.

By rapidly iterating on both **software** and **hardware** using Codasip Studio, you will witness firsthand how targeted instruction sets can deliver **substantial performance boosts** to LLM primitives.

First read the background document, which will give you some insight into the workings of Transformers. Next is the tutorial and instructions documents, which will teach you Studio and describe the provided CPU and code.

# **PREREQUISITES**

Base AI/ML understanding (MLP, CNN, activations etc.)

Base Linear Algebra understanding (vectors / matrices)

Base C-coding skills.