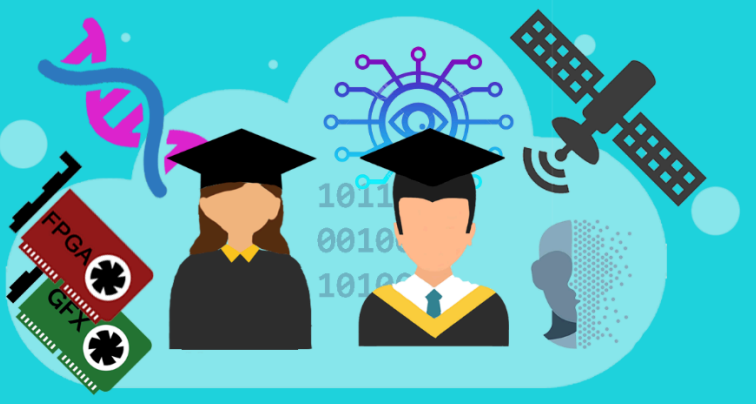
Diploma Thesis

Microproccessors and Digital Systems Laboratory



Multi-ISA LLM Inference and Deployment on the Edge

Need for LLM Inference on the Edge

Nowadays, the Large Language Models (LLMs) have been an inseparable part of our society (link), making them, if not the most, one of the most discussed research topics of recent years. The complexity of those models is exponentially increasing almost month by month, which also leads to the increase of the demands of certain computational capabilities, both on the training and inference parts of those models, along with the negative outcomes that could come out of this, which include

- Increase in **Energy** Needs
- Increase Computational **Area** (that also increases the ASIC fabrication cost)
- HW cannot keep up with the SW Demands
    - Models becoming **Compute/Memory Bound**

The increase in both the size and parameters of the given LLM models is also presented in the following figure (Figure 1).

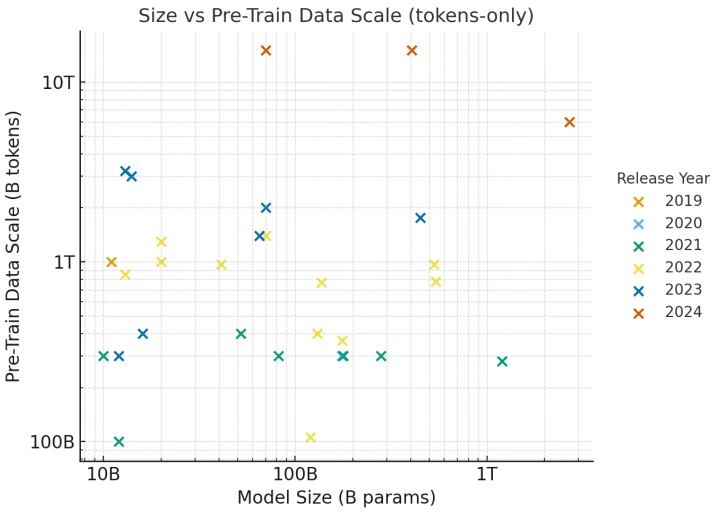


Figure 1: Scaling of the Size and Tokens on Major LLMs Presented After 2019

Along with the huge explosion of parameters, LLM inference on edge devices is sometimes necessary for various reasons, such as minimizing latency, ensuring security/privacy, achieving energy savings, having a compact constructional form factor of the accelerator, and independence from the host (either at the physical level or over the network).

Many devices and works have approached the task of model inferencing on the edge, mainly targeting how to be both energy efficient and achieve acceptable performance. Exploring this trade-off is the main spark, embracing the need for further research in this field.

Dynamic Voltage and Frequency Scaling (DVFS)

The devices, to be able to change both their energy demands and their clock speed at runtime, to change positions on the Pareto front discussed above, implemented a methodology called **DVFS** (link), which lets the user configure both the Voltage scale in which core components of the microcontroller operate (e.g. the Compute Logic, Memories, and Certain Peripherals) as well as the clock frequency that the system can operate. With this method, devices can **adapt per workload** to achieve the greatest trade-off position on the energy/speed axes mentioned above.

Of course, this task requires an extensive profiling of the workloads given, as well as proper knowledge of the computation/memory capabilities of the target device for proper (Voltage, Frequency) pairs to be selected for each case.

HW-SW Co-design for efficient DVFS and Multi-ISA Architectures

While DVFS techniques are promising for enhancing the energy efficiency of low-end systems, their usage often imposes harsh latency overheads. The imposed overheads make the use of these techniques prohibitive for real-time applications and resource-constrained devices. Frequency scaling on low-end baremetal systems often involves stalling until the clocks are reconfigured, increasing the runtime and inducing additional overhead per clock switch.

Current works have proposed decoupling the compute and memory-bound components of each workload to enhance DVFS efficiency. This relies on a simple intuition: when the workload is memory-bound, the processor stalls while “waiting” for the memory, meaning we can scale down the frequency with lower latency penalties [1][2] . Recent works employ hardware-software co-design on the workloads to uncover more efficient decoupling opportunities and clocking strategies [3], or maximize DVFS efficiency for multi-core big.LITTLE (link) architectures[4]. However, these approaches are either focused on single-core architectures [3] or exploit very coarse-grain decoupling/core offloading strategies[4], leading to inefficient execution.

Transferring those principles to Multi-ISA research, big.LITTLE architectures can be considered as a “soft” description of multi-ISA implementation. In this architecture, the computational resources are organized in high-performing (high energy) and low-performing (low energy) ARM-based cores, utilizing different combinations of them on each instance, providing also different options in the power/latency trade-offs. Although big.LITTLE delivers a combination of different architectures on a single SoC; there are more aggressive schemes of this principle (as we will see), targeting entirely different architectures on a single SoC platform.

Alongside the commercial solutions discussed above, the multi-ISA optimization problem has also been an interesting topic of research, providing impactful results in several cases (link - HPCA 2019).

Raspberry Pi Pico 2

The Raspberry Pi Pico 2 development board is based on the RP2350 SoC and provides a **Dual-Architecture principle** to the developer community, offering both 2 Arm Cortex-M33 cores and 2 RISC-V Hazard3 Cores, also including 520KB of SRAM (in 10 banks), 16MB of on-chip Flash, and support for up to 16MB of external QSPI flash/PSRAM. Along with this novel approach to SoC customization, the small form factor, as well as the low energy needs of this SoC, provide a great choice for edge devices, focusing on the ultra-low-power hardware design community. A view of the RPPico2 Development Board, as well as an abstract schematic of the driving circuit that the dual-architecture is based can be seen in Figure 2.

The RP2350 can be programmed both in C/C++ with a well-supported open-source SDK, provided for both the ARM and RISC-V compilation flows, and also provides high-level control for **DVFS** and communication with different peripherals that are supported. The SDK also has limited control over the Dual Architectures, **letting the user change the Architecture at runtime**, letting ARM or RISC-V be the operating architecture (unfortunately, there is no support at this moment for both architectures to operate at the same time from higher-level programming). Finally, the SDK also supports a Python Interpreter Runtime (microPython), but due to the research focus of this thesis, we will probably not utilize this feature.

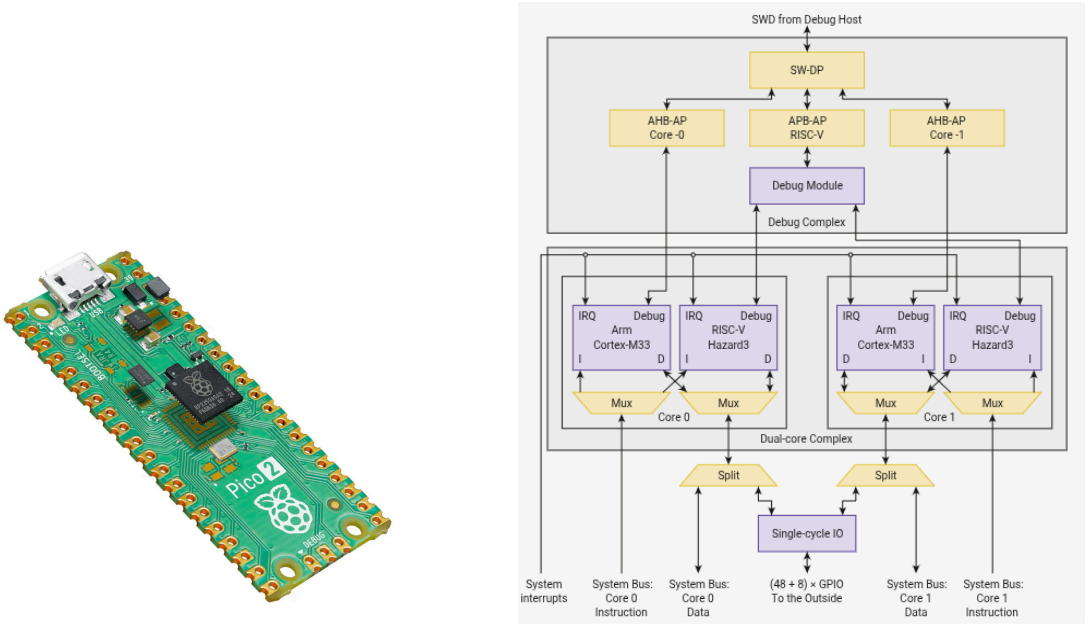


Figure 2: a) The RPPico2 Development Board, b) The Dual-Architecture Schematic of RP2350

Focus of this Thesis

In this thesis, we will focus on two separate topics, based on the Raspberry Pi Pico 2 Board. The first topic is how we can optimally utilize DVFS methodologies upon novel LLM networks, to achieve the lowest latency and highest performance trade-offs, benefiting also from the dual architecture, letting the user also decide which processor is best fitted for each workload given, producing at the end optimal (Voltage, Frequency, Architecture) pairs per workload package.

The second topic will be based on the affinity of the LLM-based workloads themselves. By affinity, we mean that the dataflow representation of each workload (e.g. the loop formation in a gemm operation) will be profiled to run optimized on the RP2350, utilizing the best dual architecture paradigm of this SoC. Those workloads after the profiling, they will choose which architecture they can perform better or consume less energy, bringing also the option for the RP2350 to **operate in dual-architecture mode**, which is an unexplored field in this community, and it will also require the developer to dig into the interesting architecture parameters of RP2350.

TOPICS AVAILABLE: 2

PREREQUISITES:

Familiarity with:

- Embedded Systems
- Programming in C and Assembly
- Basic Microcontroller Programming

Desirable qualifications:

- Basic Knowledge of LLM networks

RELATED MATERIAL:

[1] Towards more efficient execution: A decoupled access-execute approach

[2] "Fix the code. don't tweak the hardware: A new compiler approach to voltage-frequency scaling,"

[3]Decoupled Access-Execute Enabled DVFS for TinyML Deployments on STM32 Microcontrollers

[4] Fine-grained energy efficiency using per-core dvfs with an adaptive runtime system

- Leakage aware dynamic voltage scaling for real-time embedded systems

CONTACT INFORMATION:

Georgios Alexandris, Ph.D. Student, NTUA: galexandris@microlab.ntua.gr

Elli Alvanaki, Ph.D. Student, Columbia University: ealvanaki@cs.columbia.edu

Dr. Manolis Katsaragakis, Post-Doc Researcher, NTUA: mkatsaragakis@microlab.ntua.gr

Asst. Prof. Sotirios Xydis, NTUA: sxydis@microlab.ntua.gr