www.inesc-id.pt

An FPGA-based platform to evaluate Posit arithmetic in next-generation processors

Nuno Neves[1], Luís Crespo[1], Federico Rossi [2], Marco Cococcioni [2], Sergio Saponara [2], Martin Kuehn [3], Jens Krueger [3], Pedro Tomás[1], Nuno Roma[1]

[1]INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal; [2]University of Pisa, Pisa, Italy; [3] Fraunhofer ITWM, Kaiserslautern, Germany

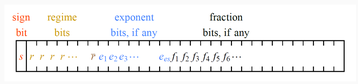
Summary

The recent Posit number system has been regarded as a particularly well-suited floating-point format to optimize the throughput and efficiency of low-precision computations. Besides offering very-low precision format configurations, where standard floating-point formats are yet to provide support, the Posit format offers a balance between decimal accuracy and dynamic range, resulting in a distribution of values that has been showing quite interesting results in selected application domains. However, when compared to other formats, the adoption of the Posit still raises some concerns regarding hardware complexity, particularly when accounting for the overheads associated with the format decoding/encoding and the quire exact accumulator. Also, while some domains (e.g., ML/AI) have shown to benefit from the Posit format, it is still necessary to assess the viability of its adoption on a much wider range of applications. The herein presented Posit Test Array (PTA) provides an FPGA SoC-based testing vehicle that allows researchers to easily deploy floating-point arithmetic unit IP cores for functional testing. A dedicated software API for quick application evaluation provides a seamless interface to offload arithmetic operations from the SoC CPU to the IPs deployed on the FPGA fabric. The PTA was deployed on a Xilinx ZCU106 board, running Petalinux-based system with a base array comprising 5 state-of-the-art Posit/IEEE-754 FMA unit architectures. The platform is currently being exploited to validate the use of the Posit format in a wider range of applications domains and to assess its integration in the cutting-edge accelerator architectures under development in EU's European Processor Initiative.

Posit Number System

The Posit floating-point format was introduced in 2017 as an alternative to the IEEE-754 standard.

A posit number is defined by the pair <n , es>, where n represents the precision and es denotes the maximum exponent size, resulting in the following binary encoding format:



The encoded decimal number is given by the following expression:
(where *S* – sign, *e* – exponent, *m* – regime run-length, *f* - fraction)

$$(-1)^S \times 2^{e+k2^{es}} \times 1.f$$

$$\text{w. } k = \begin{cases} m-1, & r=1 \\ -m, & r=0 \end{cases}$$

Main Features

- Dynamic encoding with regime filed provides native balance between dynamic range and decimal accuracy
- Only two special encodings (for 0 and Not-a-Real)
- Fused arithmetic supported by an exact accumulator *(quire)* with size 16**n* bits.

Preliminary Studies:
Dynamic Range vs. Decimal Accuracy

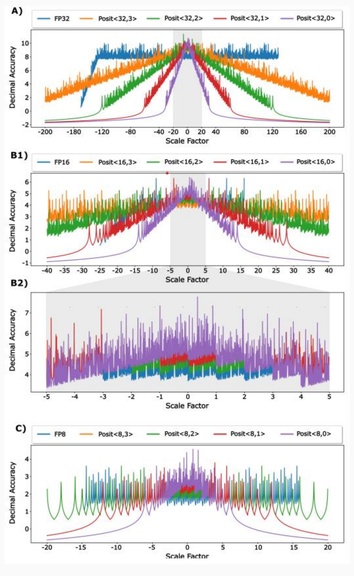
An exhaustive analysis measured the compromise between dynamic range and decimal accuracy offered by Posit standard precision, in relation to the IEEE-754 counterparts:

$$\text{Decimal Accuracy} = -\log_{10}\left|\log_{10}\left(\frac{x_{computed}}{x_{exact}}\right)\right|$$

The Posit format presents a clear tapered behaviour and evident trade-offs, when compared to the standard IEEE-754:

- Higher accuracy for numbers with a near-zero scale factor (i.e., near [-1,+1]) → ***golden zone***
- While at large precisions [**A)** 32-bit] there is a much higher accuracy at the golden zone, when the dynamic range increases it is surpassed by IEEE-754, until it is close off at zero or infinity.
- At lower precisions [**B)** 16-bit and **C)** 8-bit], by reducing decimal accuracy Posit can reach a much larger dynamic range before saturating.
- As the exponent size increases, Posit decimal accuracy flattens, resulting in a more balanced accuracy outside and inside the golden zone.

Remarks: Given the current lack of standard low-precision formats, Posit could potentially offer support for several application domains that can benefit from dedicated low-precision arithmetic.



System Overview

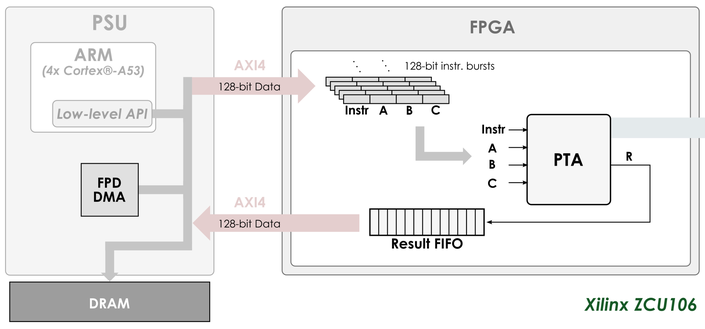
The Posit Test Array (PTA) is a robust platform that offers rapid-prototyping and testing facilities for floating-point units. It was especially devised to aid the development and evaluation of new Posit fused multiply-add (FMA) and fused multiply-accumulate (MAC) unit architectures. It was developed with the most recent Xilinx MPSoC boards, composed by tightly-coupled CPU-FPGA devices.

On the **FPGA** side, PTA is composed by:

- Test unit wrapper template, used to interface with FMA/MAC IPs.
- An AXI4-compatible configurable array capable of housing multiple test units.
- Dedicated 128-bit controller responsible for accepting, decoding, and redirecting instructions to each test unit.
- A FIFO buffer that gathers results for the PTA test units and sends them back through the AXI4 bus.

On **CPU** side, a Linux-based system deploys a dedicated API, comprising:

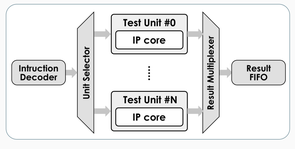
- A memory-mapped interface for direct communication and offloading to the PTA inside the FPGA.
- An arithmetic operator library that abstracts all the control and communication necessary to operate the PTA.
- DMA capabilities to improve communication throughput between the CPU and the PTA.



Arithmetic Unit Array

The PTA module is composed by an array of arithmetic units in multiple parallel lanes.

Each unit is deployed with a wrapper template that decode incoming instructions and generates the corresponding control signals according to the underlying IP specification.



Programming Interface and Linux-based API

PTA instructions and output data are sent through a 128-bit AXI4 interface memory-mapped to the SoC CPU:

- Incoming data comprises packed 128-bit words, containing a 32-bit instruction and 3 32-bit floating-point operands.
- Computing results of each unit are gathered in FIFO buffer, accessible by reading from the interface.

The control of each unit is ensured by a 32-bit instruction word, composed of:

- 4-bit opcode, encoding the operation to be performed
- 4-bit format mask, to indicate the format (Posit or IEEE-754) of each operand and from the result.
- 4x 3-bit exponent size, indicating the *es* configuration for each Posit operand.
- 8-bit reserved field, to deploy custom operations/functions dedicated to specific to the underlying unit IP.
- 4-bit unit selection of the target unit.

The PTA is managed by a dedicated API that provides C/C++ interface, in a Petalinux OS.

- Communication is handled by creating a pointer to the PTA physical address in the FPGA (using *mmap()*, through a file descriptor to */dev/mem*).
- Control instructions are generated through dedicated functions associated to each operation.

System Deployment – European Processor Initiative (EPI)

The PTA system was deployed on a Xilinx ZCU106 board, running Petalinux v2020.2. It is currently hosting multiple studies, under the EPI European project, to assess the viability of the Posit format and its underlying architectural requirements (including, resource overhead, performance, and energy efficiency), for a wide range of applications domains.

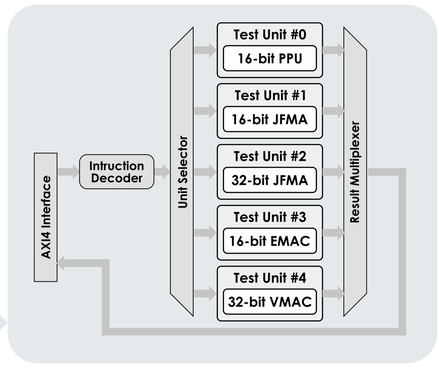
The current platform comprises an array that aggregates the following state-of-the-art Posit/IEEE-754 FMA/MAC unit architectures:

- 16-bit Posit Processing Unit (PPU) [1]
- 16-bit Joined Posit/IEEE FMA (based on [2])
- 32-bit Joined Posit/IEEE FMA (based on [2])
- 16-bit Posit MAC, with variable exponent size [3]
- 32-bit Unified Posit/IEEE Vector MAC (VMAC) [4]

Software support is provided through the **cppPosit** library:

- Template-based posit definition and configuration.
- Compile-time selection of different backends (software- or hardware-based).
- Use of operator overloading to hide backend implementation details to end-user.
- Ability to seamlessly target the PTA with operator overloading directly on the application code with just a type definition → overloaded operators call the Linux-based PTA API functions to execute arithmetic operations.

Evaluation of the different Posit format configurations and features, compared with IEEE-754 counterparts, is currently underway, to provide an extensive assessment of the adoption of the Posit format in the accelerator architectures under development for the next EPAC architecture iteration [5].



Ongoing Studies: Physics simulations

The PTA is currently being used to assess the viability of using the 16-bit Posit (vs. IEEE-754 32-bit) for Reverse Time Migration (RTM) simulations.

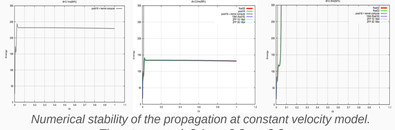
In RTM an acoustic wave equation is solved in an isotropic 3D model. A Finite Difference (FD) stencil of 4th order is used to perform the propagation.

Test Setup: An image of a single test shot in 3D is compared using different compression formats in the FD kernels. To check numerical stability the energy conservation is tested over time for different timesteps.

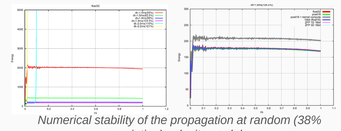
Preliminary Remarks: No or neglectable decrease in stable timesteps has been observed, indicating the potential viability of adopting 16-bit Posit for RTM simulations.

Floating-Point Format	Maximum norm	Maximum norm of difference to 32-bit IEEE-754
32-bit IEEE-754	3857	0
16-bit IEEE-754	3683	460
16-bit Bfloat	691520	691616
16-bit Posit	3687	647
16-bit Posit + kernel	4270	2356
ZFP, 1D	3869	157
ZFP, 3D	3863	21

Maximum difference of correlated images in float32 versus different compression formats.



Numerical stability of the propagation at constant velocity model. Timesteps used: 2.1ms, 2.2ms, 2.3ms.



Numerical stability of the propagation at random (38% variation) velocity model.

References

1. *M. Cococcioni, F. Rossi, E. Ruffaldi and S. Saponara, "A Lightweight Posit Processing Unit for RISC-V Processors in Deep Neural Network Applications," in IEEE Transactions on Emerging Topics in Computing, vol. 10, no. 4, pp. 1898-1908, 1 Oct.-Dec. 2022*
2. *S. Mach, F. Schuiki, F. Zaruba and L. Benini, "FPnew: An Open-Source Multiformat Floating-Point Unit Architecture for Energy-Proportional Transprecision Computing," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 29, no. 4, pp. 774-787, April 2021*
3. *N. Neves, P. Tomás and N. Roma, "Dynamic Fused Multiply-Accumulate Posit Unit with Variable Exponent Size for Low-Precision DSP Applications," 2020 IEEE Workshop on Signal Processing Systems (SiPS), Coimbra, Portugal, pp. 1-6, 2020*
4. *L. Crespo, P. Tomás, N. Roma and N. Neves, "Unified Posit/IEEE-754 Vector MAC Unit for Transprecision Computing," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 69, no. 5, pp. 2478-2482, May 2022*
5. *URL: https://www.european-processor-initiative.eu/accelerator/*

CONTACT US

nuno.neves@inesc-id.pt

Acknowledgments: This work was partially supported by Portuguese national funds through Fundação para a Ciência e a Tecnologia (FCT) under projects UIDB/50021/2020, PCIF/MPG/0051/2018, 2022.06780.PTDC, and 2022.11626.BD. It also received funding from the European High Performance Computing Joint Undertaking (JU) under Framework Partnership Agreement No 800928 and Specific Grant Agreement No 101036168 (EPI SGA2), and under Grant Agreement No 956831 (TextaRossa). The JU receives support from the European Union's Horizon 2020 research and innovation programme and from Croatia, France, Germany, Greece, Italy, Netherlands, Portugal, Spain, Sweden, and Switzerland.