PERCIVAL, the Computer That Will Boost Computing Power Using a Different Binary Code

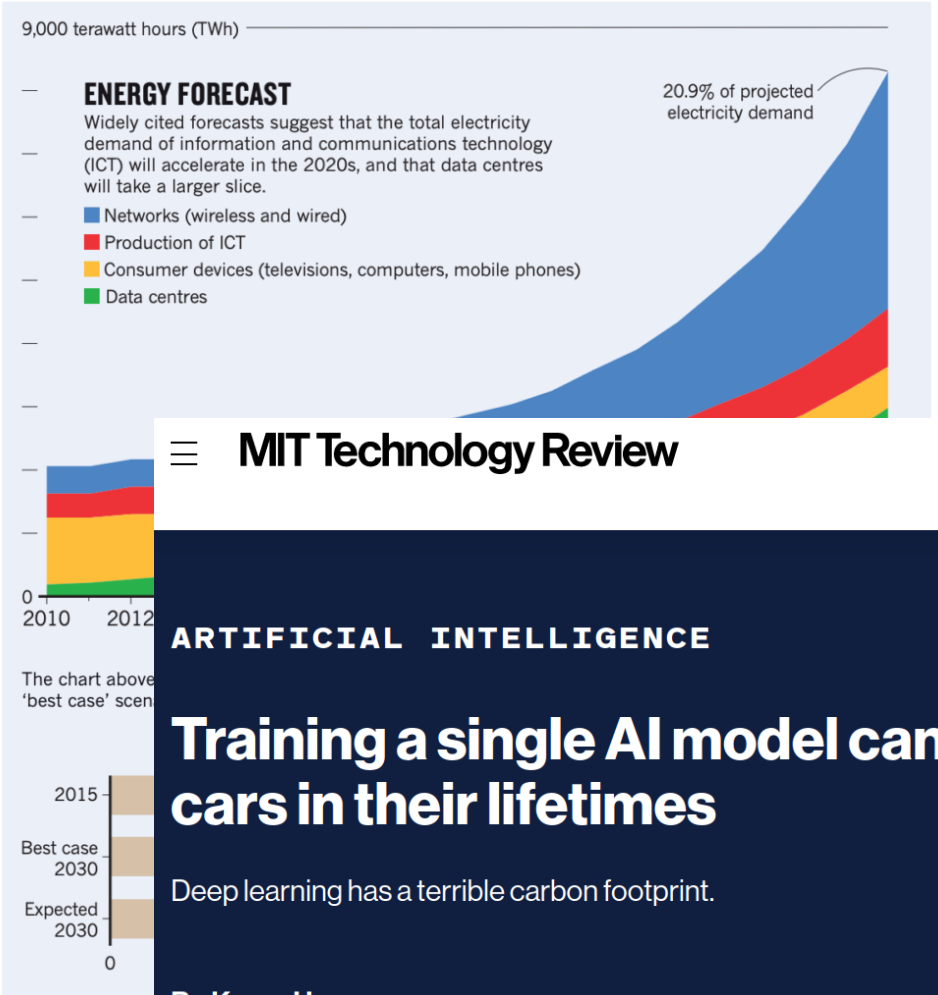
David Mallasén Quintana
dmallasen@ucm.es

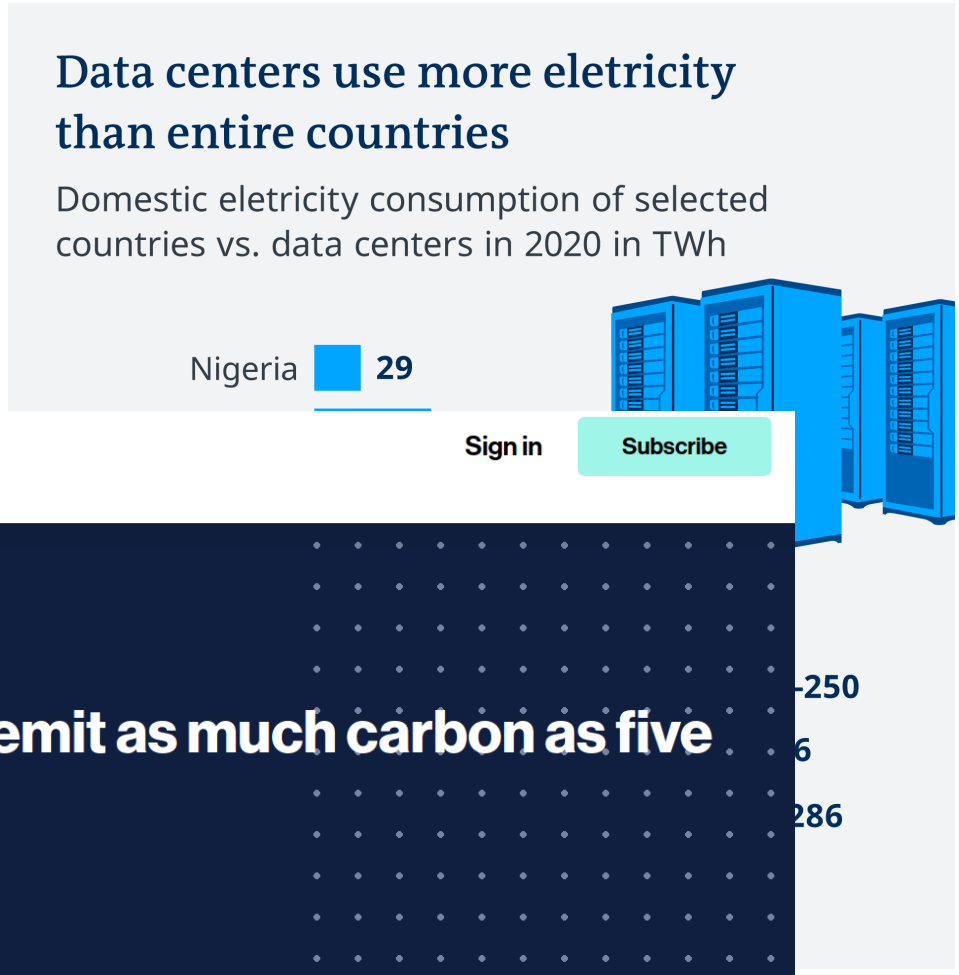
Department of Computer Architecture and Automation
Computer Science and Engineering Faculty

UNIVERSIDAD COMPLUTENSE MADRID

March 23, 2023

Motivation





MIT Technology Review

Sign in Subscribe

ARTIFICIAL INTELLIGENCE

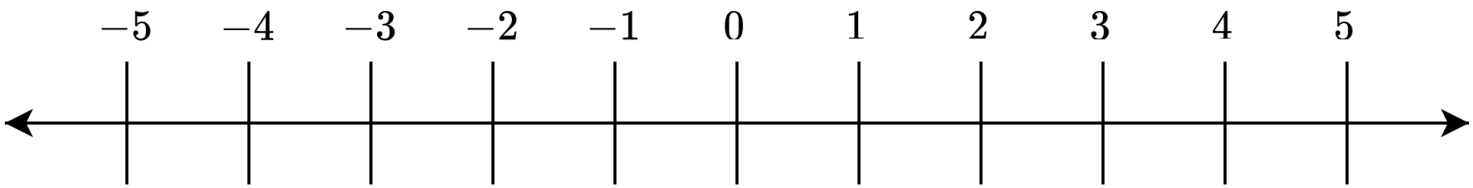
Training a single AI model can emit as much carbon as five cars in their lifetimes

Deep learning has a terrible carbon footprint.

By Karen Hao

June 6, 2019

Integer Arithmetic



- Binary representation.
- Exact precision within a range.
- With 3 bits: $000 \to 111$ can be interpreted as $0 \to 7$ or $-4 \to 3$.

$000 = 0$	$000 = 0$
$001 = 1$	$001 = 1$
$010 = 2$	$010 = 2$
$011 = 3$	$011 = 3$
$100 = 4$	$100 = -4$
$101 = 5$	$101 = -3$
$110 = 6$	$110 = -2$
$111 = 7$	$111 = -1$

- Arbitrary.
- Efficient representation.
- Decoding and encoding.

Real-Number Arithmetic

- How do we represent 1.5, $\sqrt{2}$ or π?
- Used in scientific computing, simulations, graphics...

Dynamic range

Very large numbers, e.g. Avogadro's constant $N_A = 6.022 \times 10^{23}$
Very small numbers, e.g. Planck's constant $h = 6.626 \times 10^{-34} J \cdot s$

Accuracy

There are infinite real numbers in any given range (cardinality of infinity).
We only have a finite number of bits to represent them.

IEEE 754 Floating-Point Representation

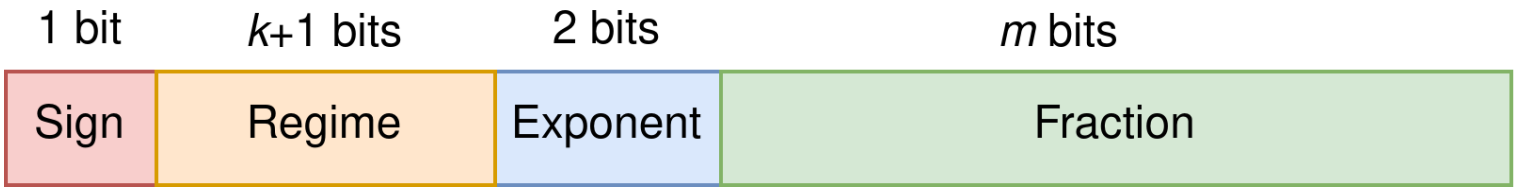
- Most widely used format.
- Established in 1985, latest revision from 2019.

Decimal value

$$(-1)^{Sign} \times 1.Fraction \times 2^{Exponent-Bias}$$

Posit Representation

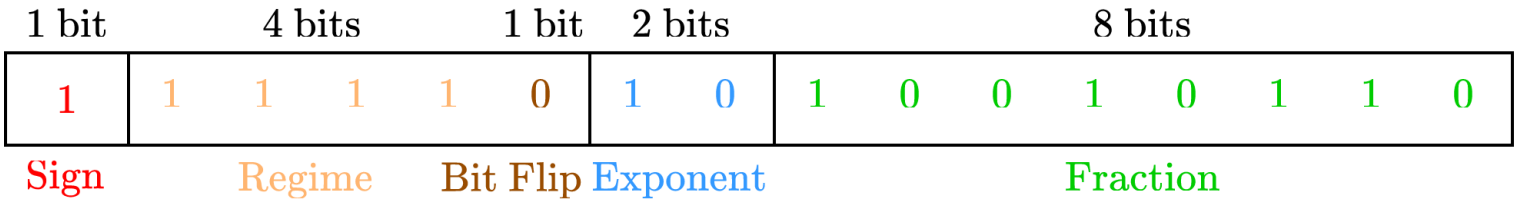
- New variable-length field: Regime.
- Variable-length fraction.



Decimal value

$$((1 - 3Sign) + Fraction) \times 2^{(-1)^{Sign}(4Regime+Exponent+Sign)}$$

Posit16 Decoding Example



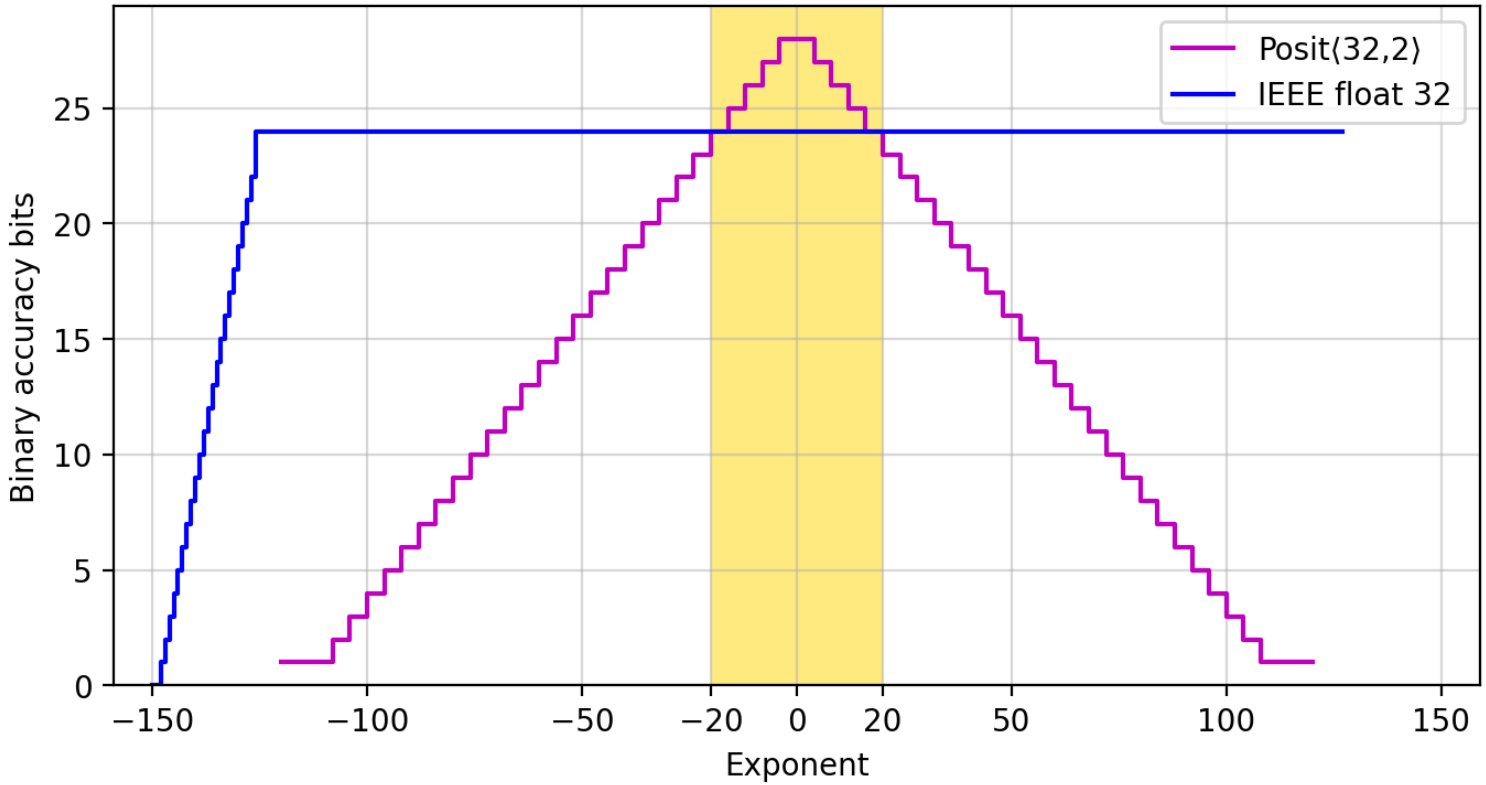
$$p = ((1 - 3s) + f) \times 2^{(1-2s)\times(4r+e+s)}, \quad r = \begin{cases} -k & \text{if } R_0 = 0 \\ k - 1 & \text{if } R_0 = 1 \end{cases}$$

$$p = ((1 - 3 \cdot 1) + \frac{150}{2^8}) \times 2^{(1-2\cdot 1)\times(4\cdot 3+2+1)}, \quad r = 4 - 1 = 3$$

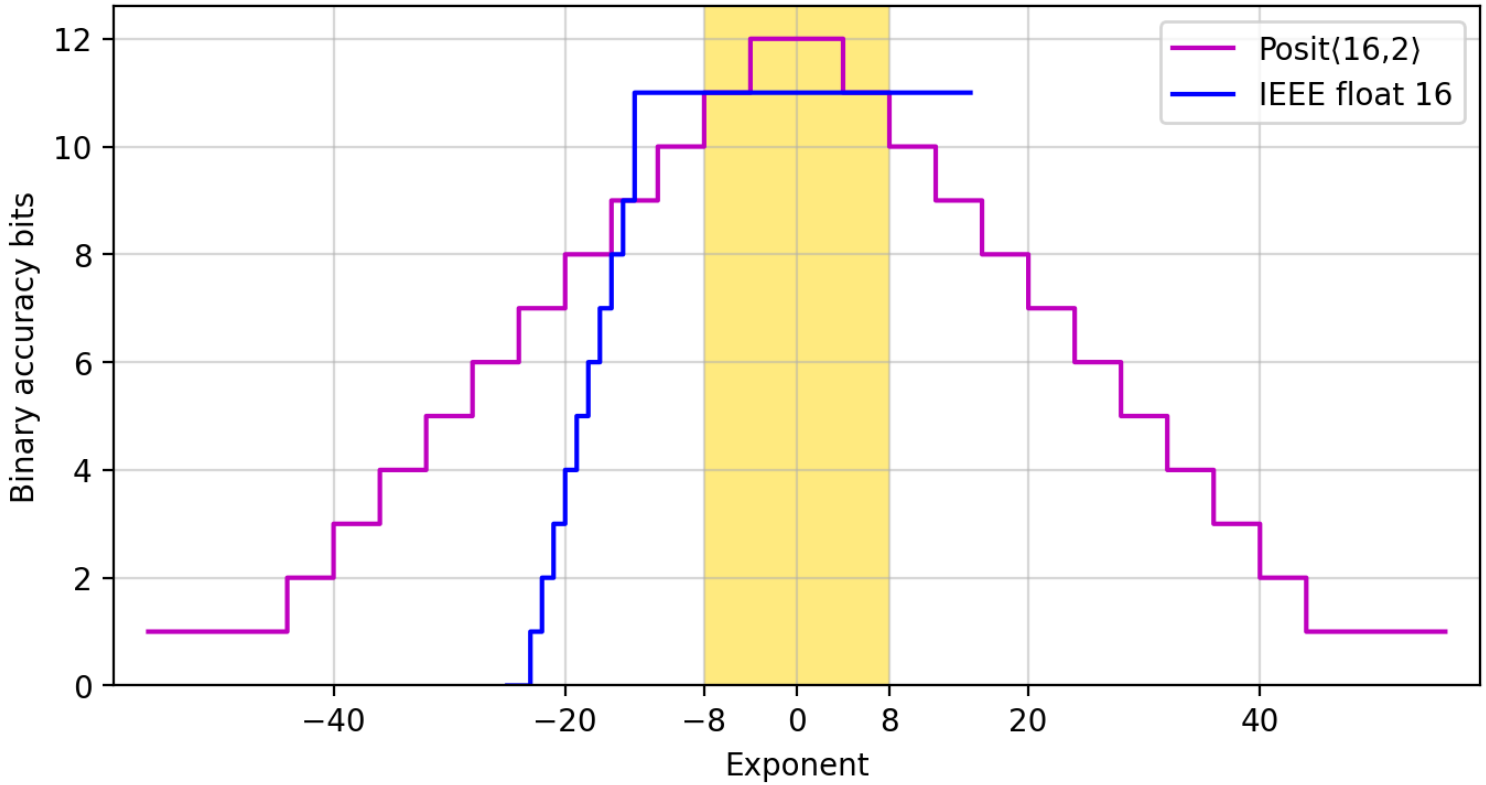
$$p = -0.000043154$$

Murillo, R., Mallasén, D., Del Barrio, A.A., Botella, G. (2022). Comparing Different Decodings for Posit Arithmetic. In: Next Generation Arithmetic. CoNGA 2022. Lecture Notes in Computer Science, vol 13253. Springer, Cham. https://doi.org/10.1007/978-3-031-09779-9_6

32-bit Floating Point vs. Posit



16-bit Floating Point vs. Posit

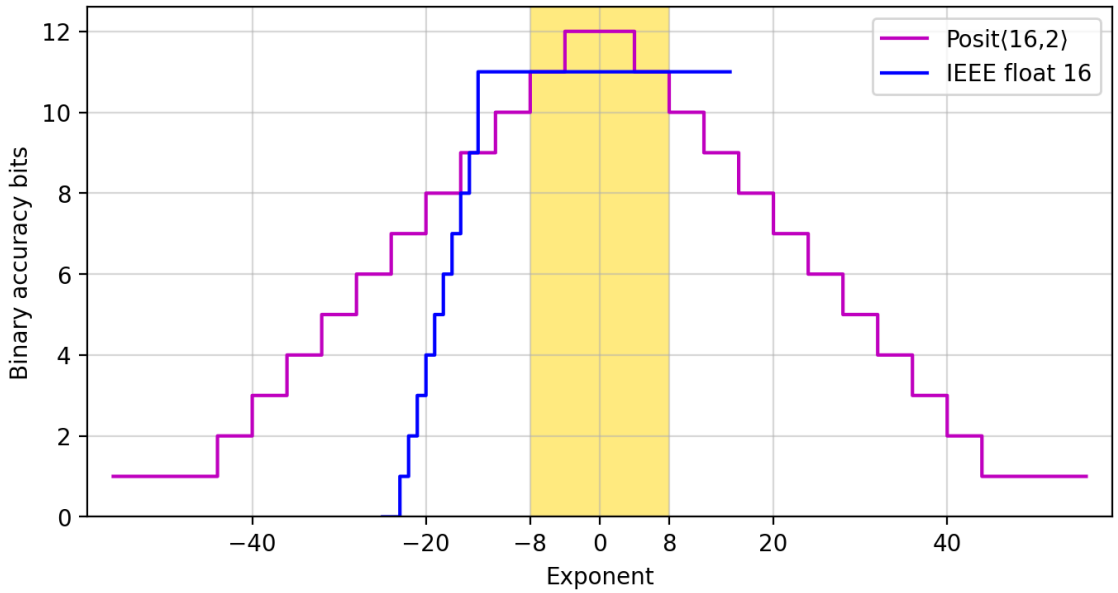


Accuracy–Dynamic Range Tradeoff

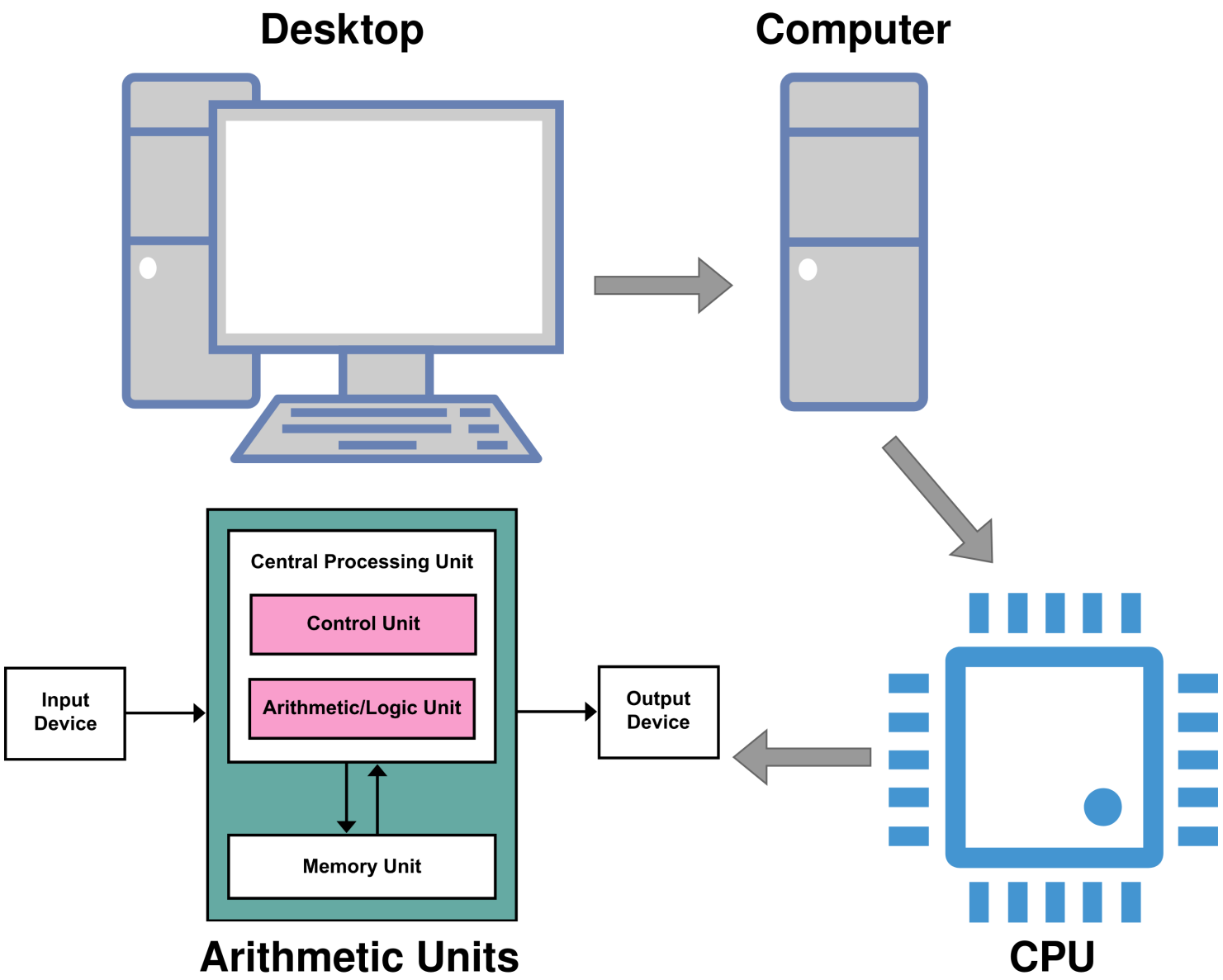
- More accuracy in a range around ± 1.
- Many calculations normalize their input values to an interval. E.g. $[-1, 1]$.
- Gradual decay of accuracy.
- Relative error: $1.1 - 1.2$ or $1000000.1 - 1000000.2$.

Avoiding accuracy loss

Always bear in mind the distribution of the values.



Inside a Computer



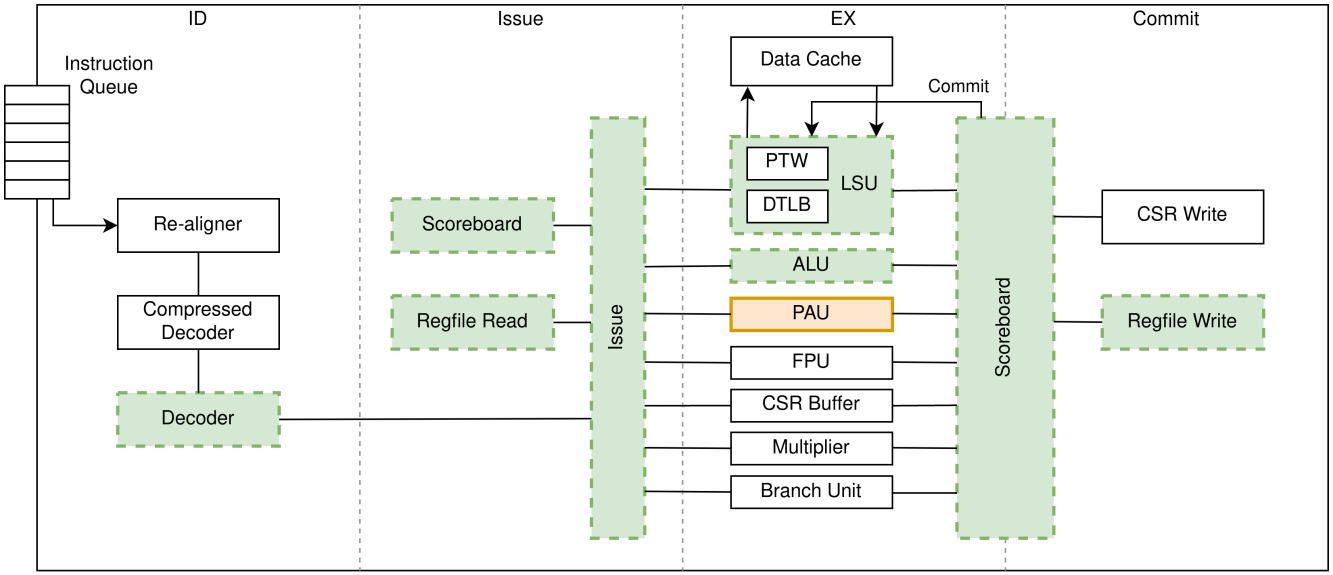
PERCIVAL Posit CPU

- Based on CVA6 developed under the PULP Platform and now maintained by the OpenHW Group.

Open Hardware!

Based on an open-source design, we also give back to the community.
`https://github.com/artecs-group/PERCIVAL`

- Application-level processor that can run a Linux Operating System.
- Adds support for all posit operations in parallel to the common floating-point.



PERCIVAL Pros and Cons

Pros

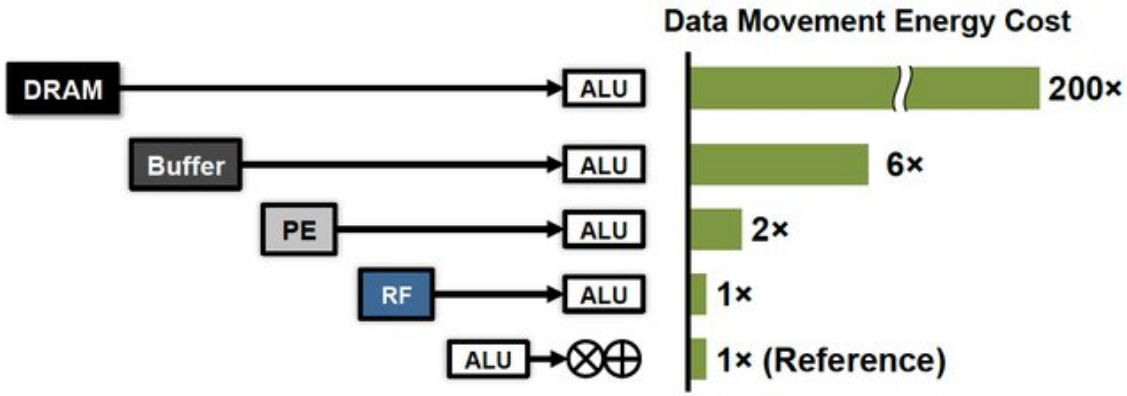
- Significantly reduce the errors of many computations.
- Lower bandwidth requirements with the use of smaller posits.
- Reproducible results. Maintain the associative property.
- Faster convergence of iterative algorithms.
- No performance degradation of executing NaN routines.
- Maintain the speed of floating-point numbers.

Cons

- Increased area requirements.
- Increased power consumption at the unit design.

D. Mallasén, R. Murillo, A. A. D. Barrio, G. Botella, L. Piñuel, and M. Prieto-Matias, "PERCIVAL: Open-Source Posit RISC-V Core With Quire Capability," IEEE Transactions on Emerging Topics in Computing, vol. 10, no. 3, pp. 1241–1252, 2022, doi: 10.1109/TETC.2022.3187199

Energy Efficiency



	DRAM	Global Buffer (>100kB)	Array (inter-PE) (1-2mm)	RF (0.5kB)
Norm. Energy	200×	6×	2×	1×

Table IV
NORMALIZED ENERGY COST RELATIVE TO A MAC OPERATION EXTRACTED FROM A COMMERCIAL 65NM PROCESS.

Smart use of energy resources

Theoretically, a n-bit posit can obtain similar precision to a 2n-bit float.
Crucial concern both for data centers and Internet-of-Things (IoT) devices.

Shi, Feng & Li, Haochen & Gao, Yuhe & Kuschner, Benjamin & Zhu, Song-Chun. (2018). Sparse Winograd Convolutional neural networks on small-scale systolic arrays.
Y. -H. Chen, J. Emer and V. Sze, "Eyeriss: A Spatial Architecture for Energy-Efficient Dataflow for Convolutional Neural Networks," 2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA), Seoul, Korea (South), 2016, pp. 367-379, doi: 10.1109/ISCA.2016.40.

Current Research

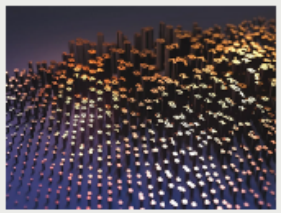
- Developing specific hardware (cores and accelerators).
- Comparisons IEEE 754 vs. Posit.
- Use of posits in scientific computing.
- Integrating our accelerators on System-on-Chip (SoC) designs.
  - Real-life applications: DNNs, biomedical, image processing, signal processing...

In the Media – IEEE Spectrum

IEEE Spectrum

NEWS | ARTIFICIAL INTELLIGENCE

Posits, a New Kind of Number, Improves the Math of AI › The first posit-based processor core gave a ten-thousandfold accuracy boost



NEWS | ARTIFICIAL INTELLIGENCE

Machine Learning's New Math › New number formats and basic computations emerge to speed up AI training

In the Media – CoNGA

John Gustafson, the inventor of posits, in his welcome speech at CoNGA 2023.

In the Media – Specialized blogs & Twitter

hackster.io
AN AVNET COMMUNITY

Search Projects

Projects Channels News Contests Events Videos

Researchers Build a RISC-V Chip That Calculates in Posits, Boosting Accuracy for ML Workloads

Designed as an alternative to floating-point numbers, posits may prove key to boosting machine learning performance.

Gareth Halfacree Follow

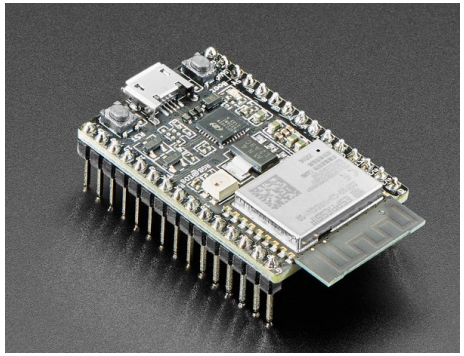
5 months ago • Machine Learning & AI

PULP Platform
@pulp_platform

It's really great to see PERCIVAL by @DavidMallasen, an application-level posit RISC-V core based on Ariane CVA6. PERCIVAL can execute all posit instructions, including the quire fused operations. ieeexplore.ieee.org/abstract/docum... Have a great weekend!

Impact

- Improving energy efficiency.
  - Important both in data centers and edge devices.
- Clear theoretical advantages in low-precision scenarios:
  - This is the current trend in Internet-of-Things (IoT).
  - Sensors, biomedical devices, wearables...
- Artificial intelligence?
  - Small posits adapt to the distribution of neural networks.
  - However, AI is very tolerant of approximations.
  - Current trend in AI inference is small integers.
  - Maybe posits could accelerate training? Still much work to do.



Design Ease-of-Use

- Our hardware units are meant to be used in any customizable CPU.
  - The targets are both FPGA and custom ASIC designs.
- The Posit Arithmetic Unit in PERCIVAL is tightly coupled inside the CPU.
- We are working on a modular solution.
  - Abstracting the need to develop control units, modify the datapath...
- Expert knowledge in hardware design and computer architecture is needed.

Compiler support

We modified the LLVM compiler adding Xposit, our custom RISC-V extension. Allows to compile C programs together with posit instructions.

Also Open Source!

`https://github.com/artecs-group/llvm-xposit`

Many open opportunities

At the hardware level:

- Increase the performance of posit CPUs.
  - Multi-core designs.
  - Integrate our units into out-of-order CPUs.
  - Posit vector accelerators.
- Modular designs that can be integrated into new CPU architectures.
- Optimization of posit hardware units.

At the software level:

- Porting new applications to leverage posit arithmetic.
  - Scientific computing, High-Performance Computing (HPC)...
  - Embedded applications, edge devices...
- Compiler C language front-end.

Contact us

We are open for collaborations!

PERCIVAL, the Computer That Will Boost Computing Power Using a Different Binary Code

David Mallasén Quintana
dmallasen@ucm.es

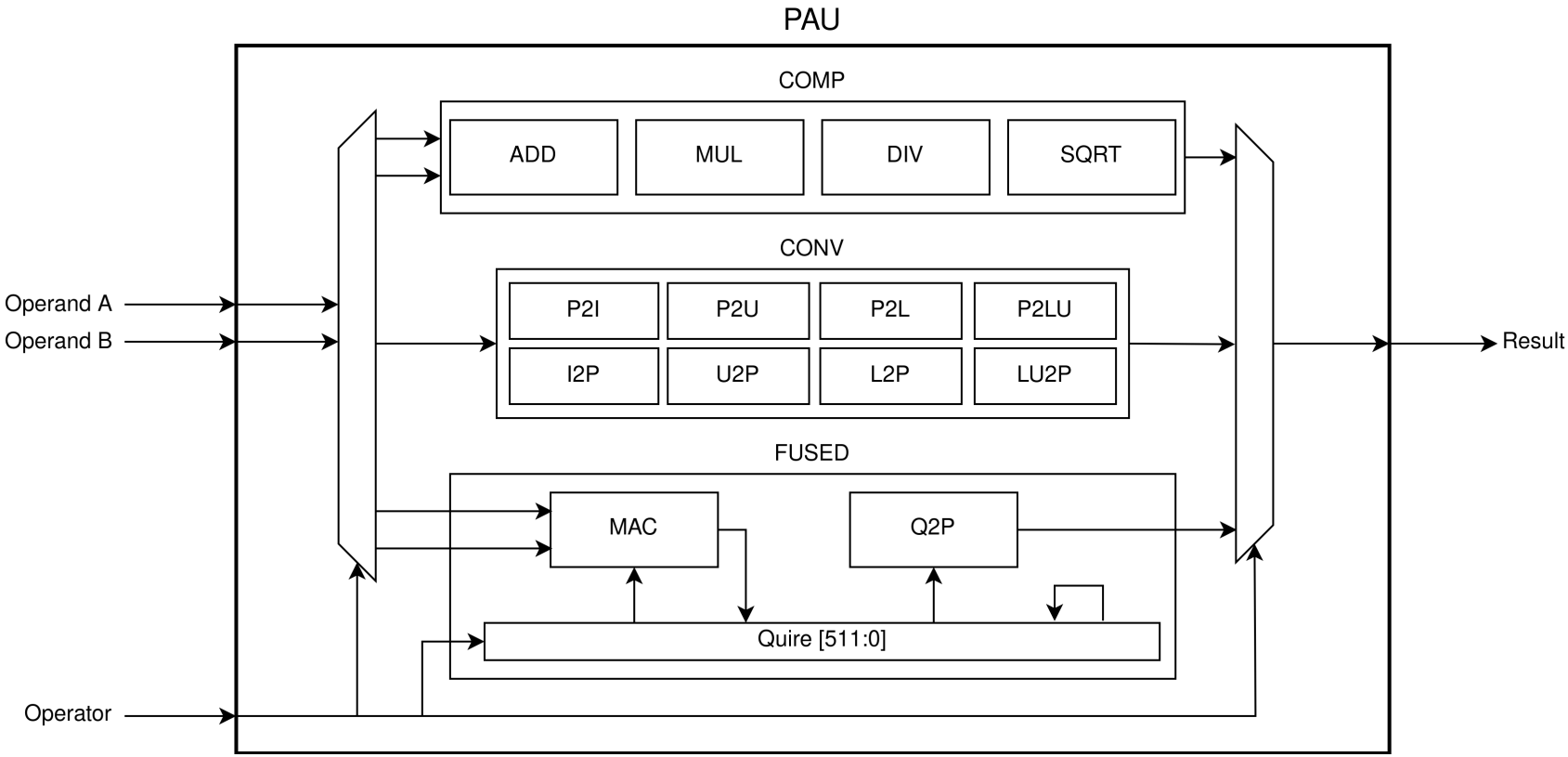
https://github.com/artecs-group/PERCIVAL

Department of Computer Architecture and Automation
Computer Science and Engineering Faculty

UNIVERSIDAD COMPLUTENSE MADRID

Backup slides

PERCIVAL Posit Arithmetic Unit



Quire Accumulator Register

- Fixed-point accumulator: for posit32 $\rightarrow$ 512-bit quire.
- $2^{31} - 1$ fused MAC operations without accuracy loss.
- Workflow of operations:
  - `qclr` : $quire \leftarrow 0$
  - `qmadd` : $quire \leftarrow quire + (posit_1 \times posit_2)$
  - `qround` : $posit \leftarrow \mathsf{round}(quire)$

  Dot product: $\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$
  Matrix multiply: $c_{ij} = a_{i1} b_{1j} + a_{i2} b_{2j} + \cdots + a_{in} b_{nj}$

Xposit RISC-V Custom Extension

31 27	26 25	24 20	19 15	14 12	11 7	6 0	
imm[11:0]			rs1	001	rd	0001011	PLW
imm[11:5]		rs2	rs1	011	imm[4:0]	0001011	PSW
00000	10	rs2	rs1	000	rd	0001011	PADD.S
00001	10	rs2	rs1	000	rd	0001011	PSUB.S
00010	10	rs2	rs1	000	rd	0001011	PMUL.S
00011	10	rs2	rs1	000	rd	0001011	PDIV.S
00100	10	rs2	rs1	000	rd	0001011	PMIN.S
00101	10	rs2	rs1	000	rd	0001011	PMAX.S
00110	10	00000	rs1	000	rd	0001011	PSQRT.S
00111	10	rs2	rs1	000	00000	0001011	QMADD.S
01000	10	rs2	rs1	000	00000	0001011	QMSUB.S
01001	10	00000	00000	000	00000	0001011	QCLR.S
01010	10	00000	00000	000	00000	0001011	QNEG.S
01011	10	00000	00000	000	rd	0001011	QROUND.S
01100	10	00000	rs1	000	rd	0001011	PCVT.W.S
01101	10	00000	rs1	000	rd	0001011	PCVT.WU.S
01110	10	00000	rs1	000	rd	0001011	PCVT.L.S
01111	10	00000	rs1	000	rd	0001011	PCVT.LU.S
10000	10	00000	rs1	000	rd	0001011	PCVT.S.W
10001	10	00000	rs1	000	rd	0001011	PCVT.S.WU
10010	10	00000	rs1	000	rd	0001011	PCVT.S.L
10011	10	00000	rs1	000	rd	0001011	PCVT.S.LU
10100	10	rs2	rs1	000	rd	0001011	PSGNJ.S
10101	10	rs2	rs1	000	rd	0001011	PSGNJN.S
10110	10	rs2	rs1	000	rd	0001011	PSGNJX.S
10111	10	00000	rs1	000	rd	0001011	PMV.X.W
11000	10	00000	rs1	000	rd	0001011	PMV.W.X
11001	10	rs2	rs1	000	rd	0001011	PEQ.S
11010	10	rs2	rs1	000	rd	0001011	PLT.S
11011	10	rs2	rs1	000	rd	0001011	PLE.S