Posit Arithmetic Units for Deep Neural Networks

Jornadas SARTECO 2021
Workshop CABAHLA

Raul Murillo, Alberto A. del Barrio & Guillermo Botella

UNIVERSIDAD COMPLUTENSE MADRID

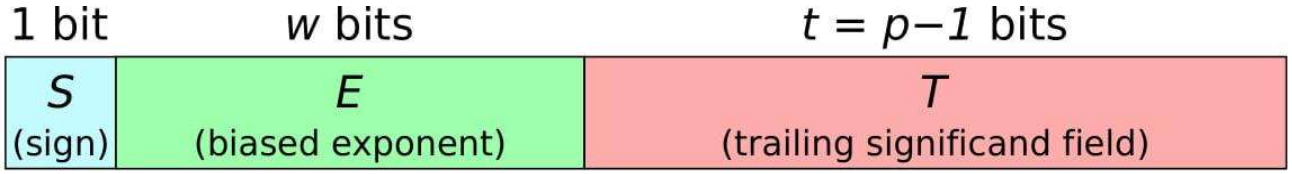
Outline of Presentation

- Introduction
- The Posit Number System
  - Properties & disadvantages
- Design of Posit Functional Units
  - Adder
  - Multiplier
  - MAC
- Using Posits in Deep Neural Networks
  - DNN training
  - Low precision inference

Introduction

Floating-point arithmetic

- Used to represent real numbers in computers
- Similar as scientific notation



$$v = (-1)^S \times 2^{E-bias} \times (1 + 2^{1-p} \times T)$$

- IEEE 754 standard
  - Implemented in the vast majority of modern computers

Shortcomings of IEEE 754

- Multiple bit patterns are wasted (NaN exceptions)
- Different results across different computers
  - Multiple rounding schemes
- Not well-suited for AI applications
  - Companies develop their own formats



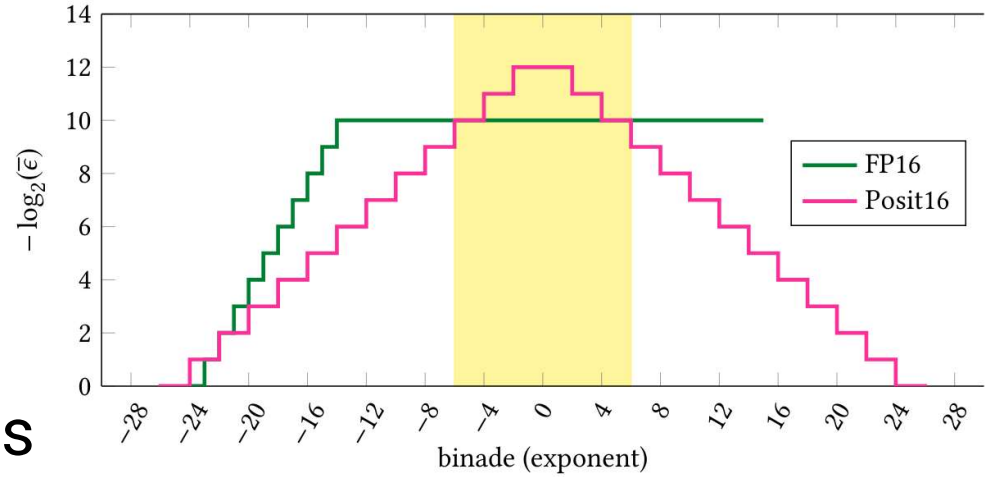
The Posit Number System

Posit arithmetic

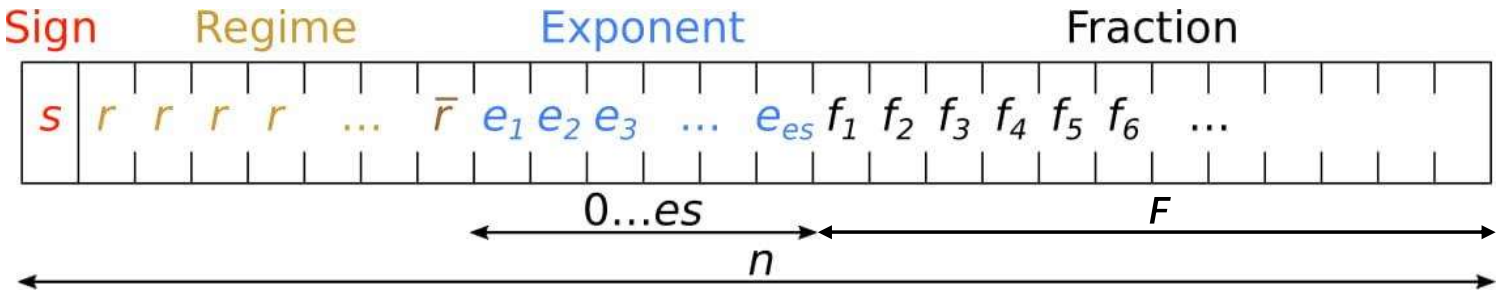
"Designed as a direct drop-in replacement for IEEE standard 754 for floating-point numbers"

John L. Gustafson

- Advantages over floats
  - Larger dynamic range
  - Higher accuracy
  - Consistency between machines
- No fixed-size fields



The Posit$\langle n, es \rangle$ format

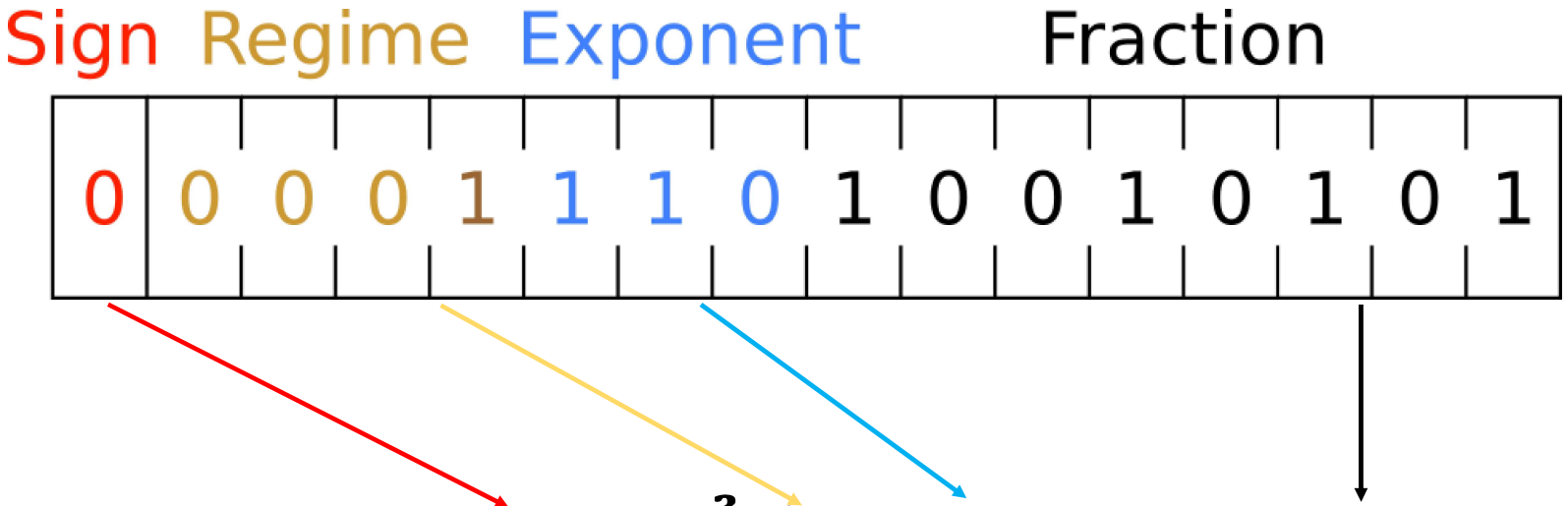


$$v = (-1)^s \times useed^k \times 2^e \times (1 + f); \quad useed = 2^{2^{es}}$$

- Sign bit (s)
- Regime (k) – sequence of m identical bits (r)

  $k = -m$ if r is 0, and $k = m - 1$ if r is 1; $m = \#$ occurrences of r
- Exponent (e) – represented by es bits
- Fraction (f) – unsigned integer divided by 2^F, so $0 \leq f < 1$

Example: Posit⟨16,3⟩

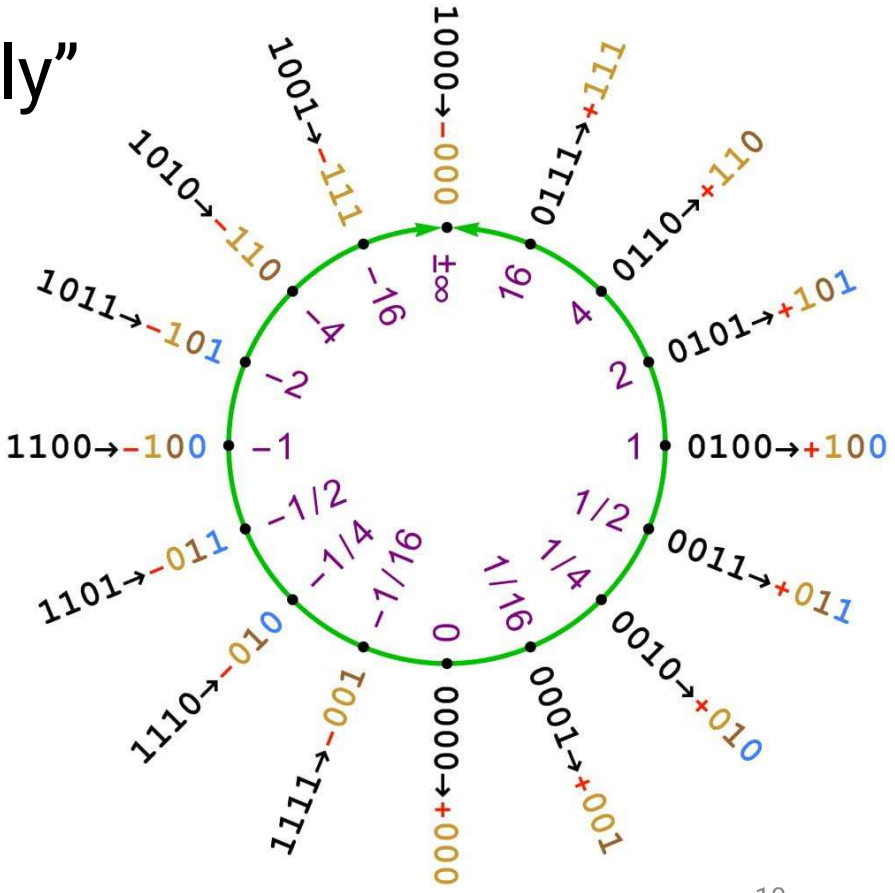


$$v = (-1)^{0} \times (2^{2^3})^{-3+6} \times 2^{2} \times (1 + 169/256)$$

$$= 6.034970283508301 \times 10^{-6}$$

Properties of posit arithmetic

- Designed to be “hardware-friendly”
- No wasted “NaN” patterns
- Single 0 and ±∞ (NaR) values
- Opposite: 2’s complement
- Comparison as integers
- Fused operations
  (*quire* register)



Drawbacks of the posit format

- Short time of life
  - Lack of tools & HW support
- Regime field (run-time variable-length)
- Worse than floats for certain domains
  - Particle physics simulation [1]

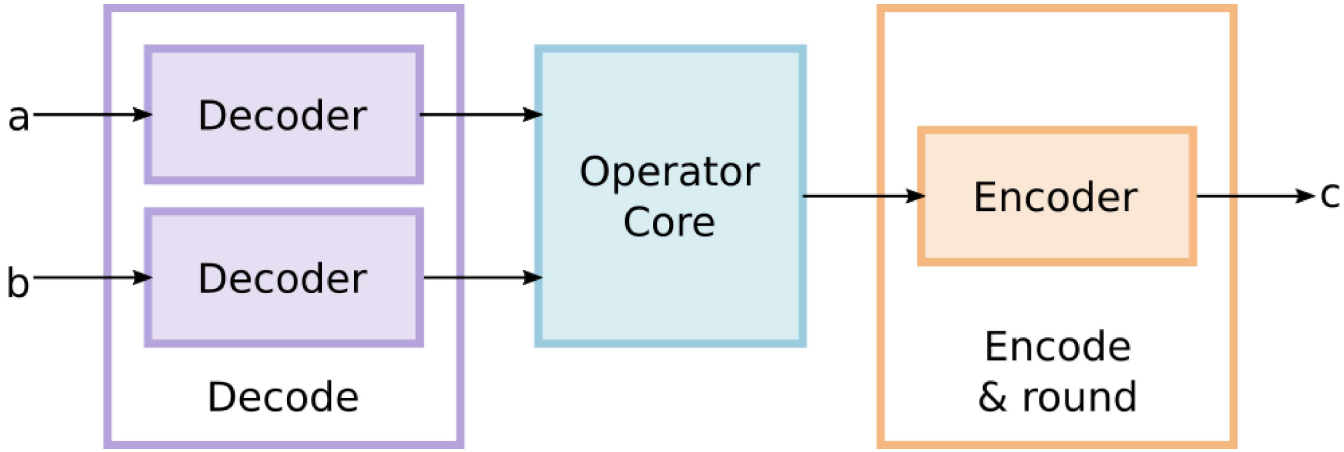
[1] F. De Dinechin et al., "Posits: the good, the bad and the ugly." *Proceedings of the Conference for Next Generation Arithmetic 2019*. 2019.

Design of Posit Functional Units

[2] R. Murillo, A. A. Del Barrio, and G. Botella, “Customized Posit Adders and Multipliers using the FloPoCo Core Generator”, in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, IEEE, 2020, pp. 1–5. DOI: 10.1109/iscas45731.2020.9180771.

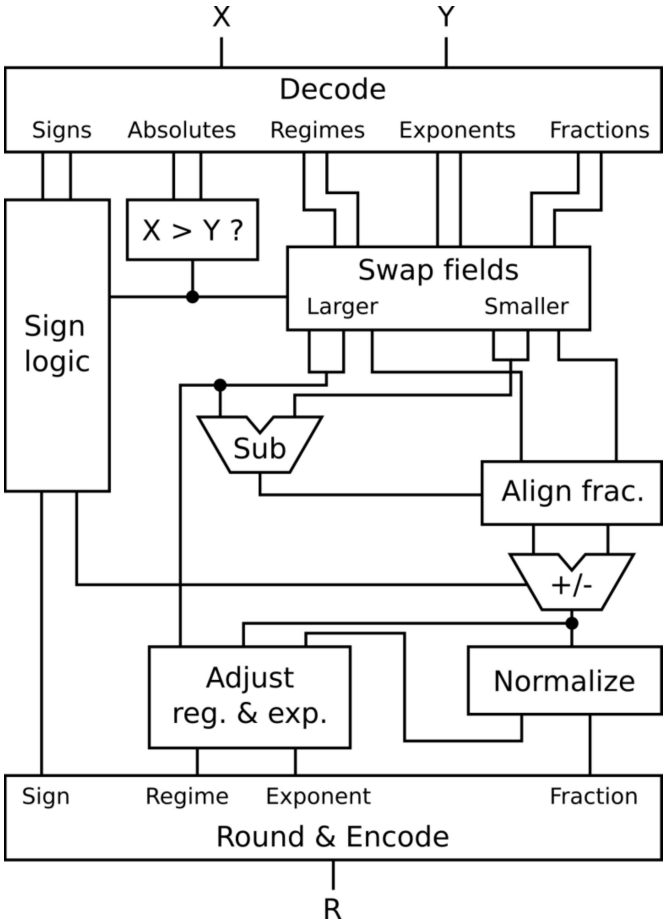
Posit functional units

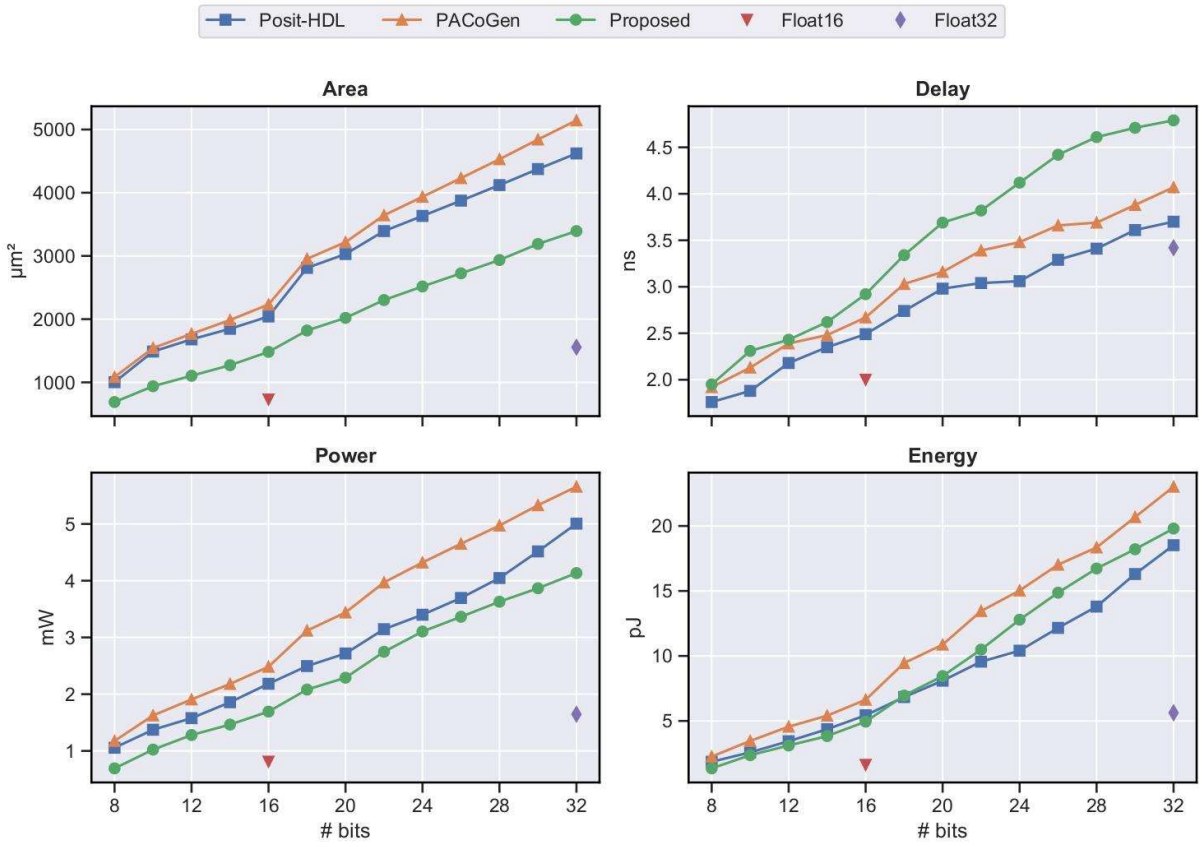
- Posits were designed to be “hardware-friendly”
  - Similar circuitry to floating point
  - Less special cases (just 0 and NaR)



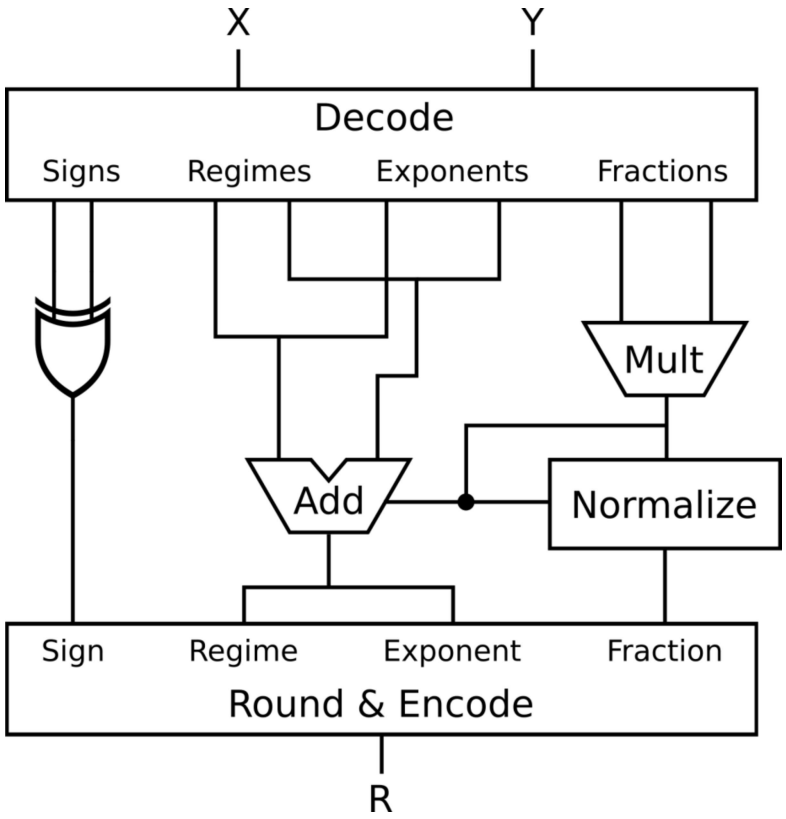
- Design challenge: runtime varying fields

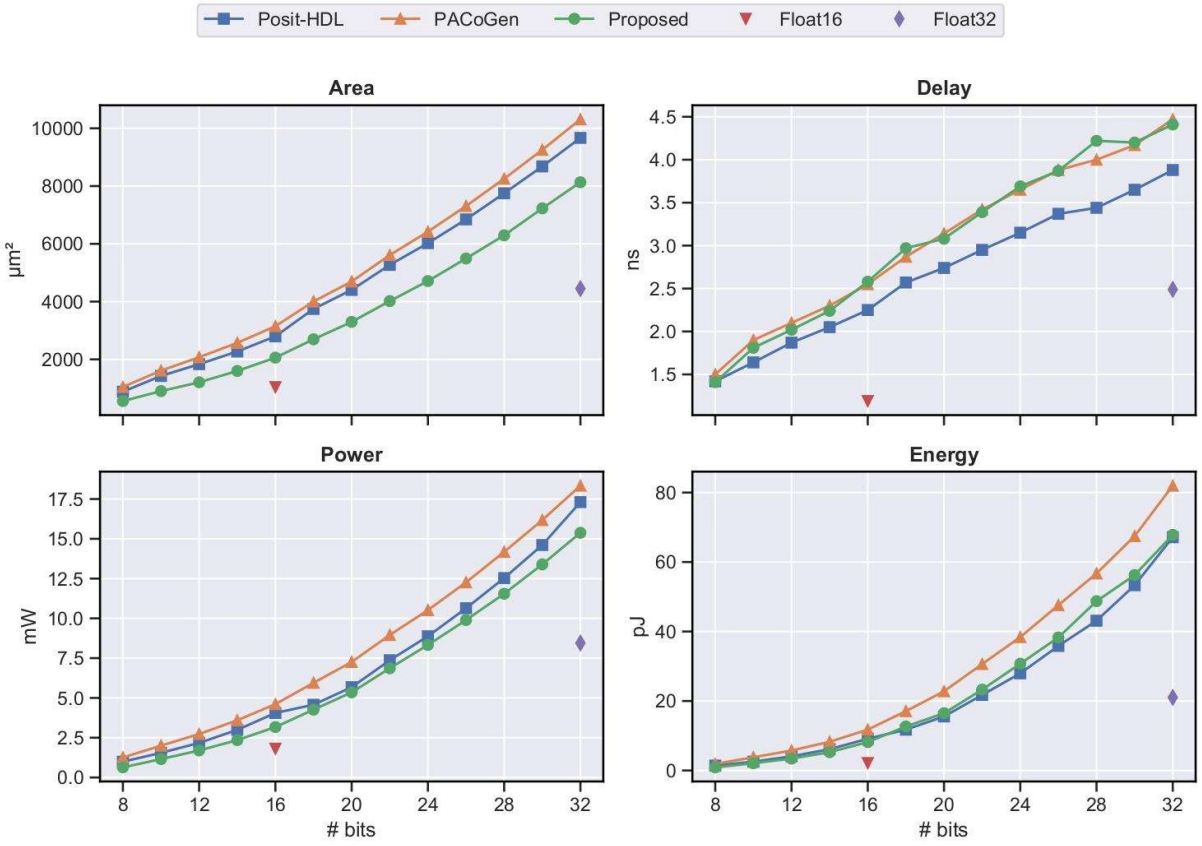
Posit functional units – Adder



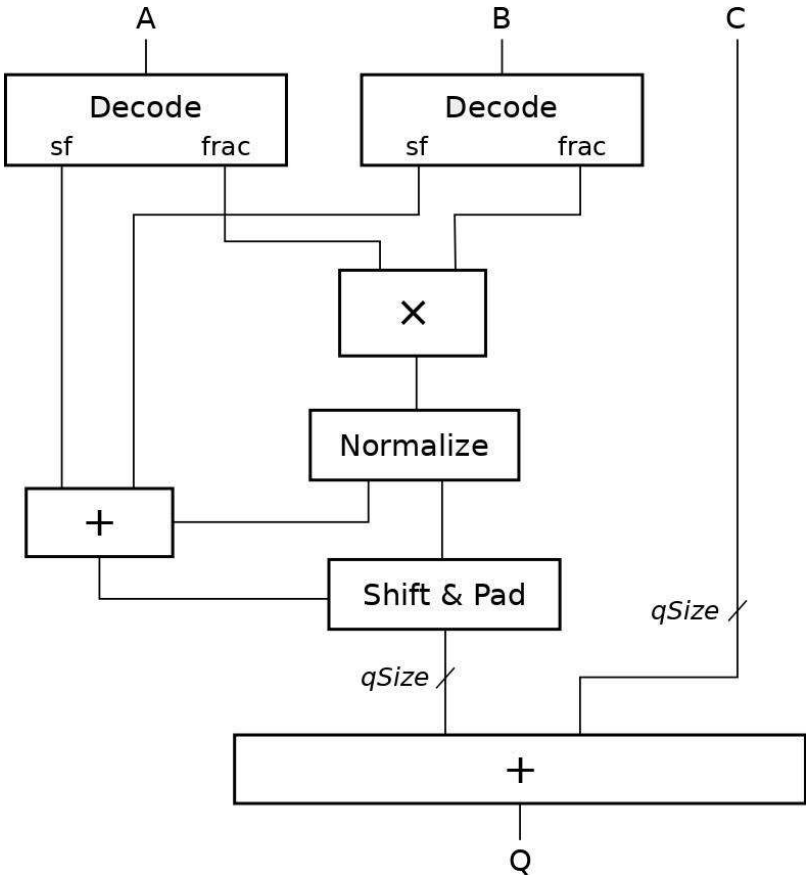


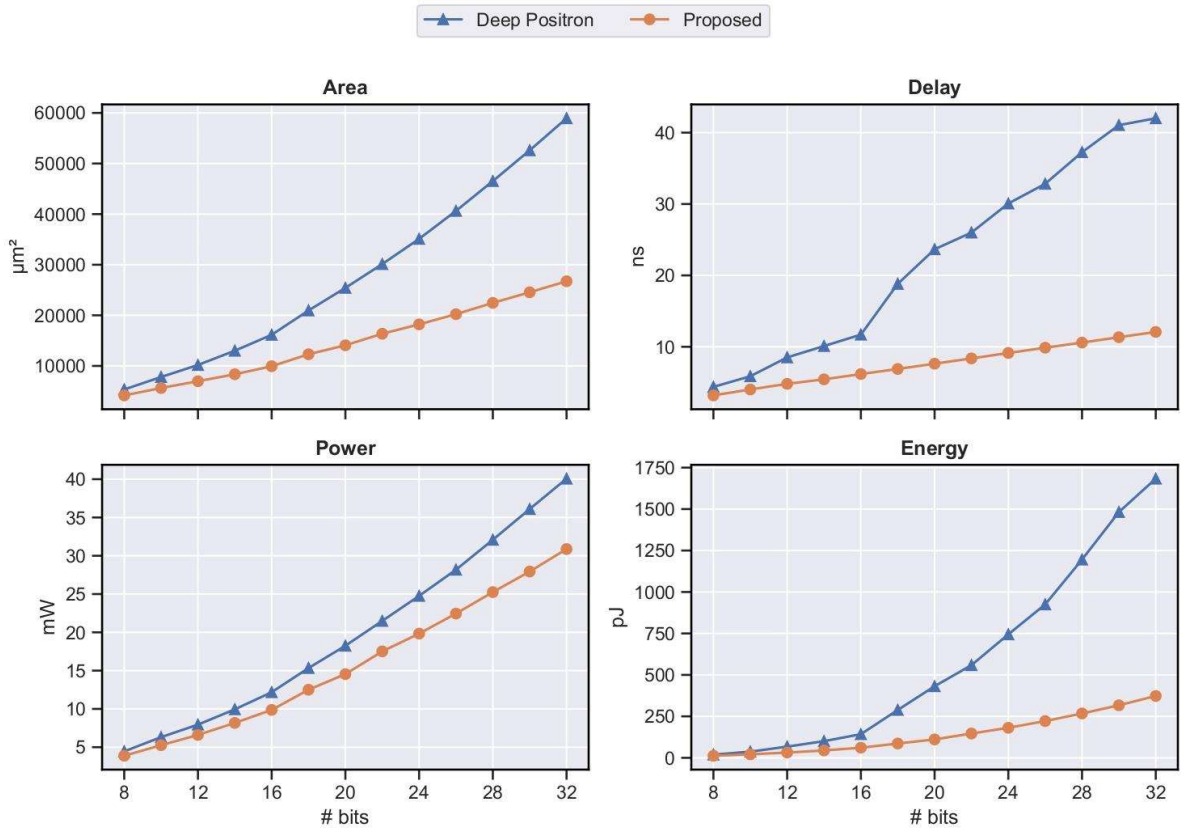
Posit functional units – Multiplier





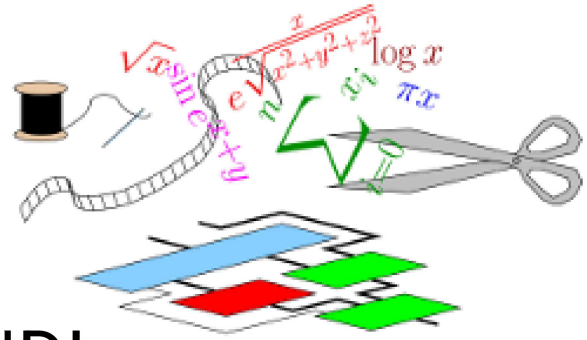
Posit functional units – MAC

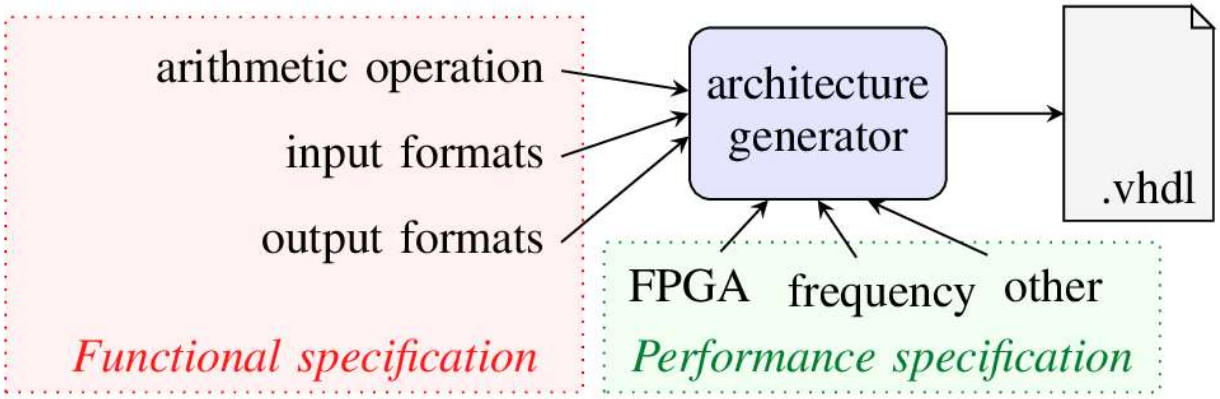




FloPoCo core generator

- Open-source tool for generating arithmetic cores for FPGAs
- Operators are fully parameterized
- Written in C++, outputs synthesizable VHDL





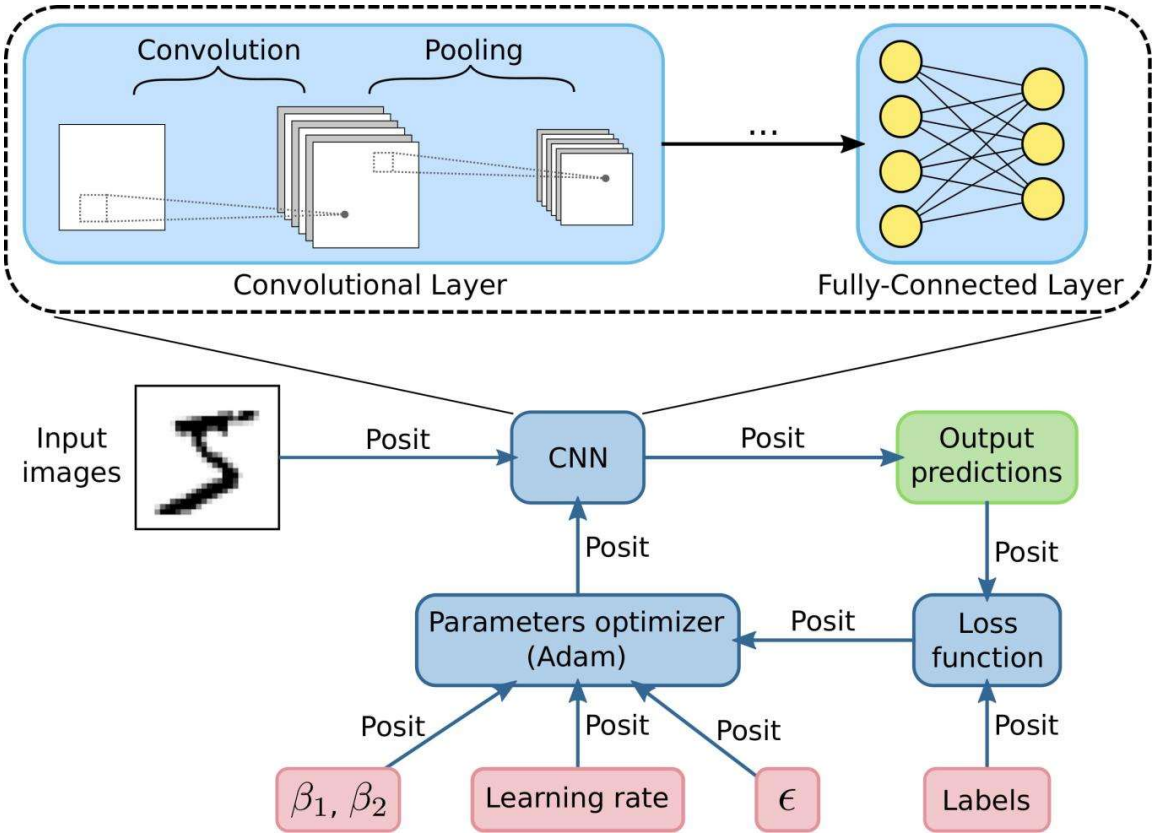
github.com/RaulMurillo/Flo-Posit

Using Posits in Deep Neural Networks

[3] R. Murillo, A. A. Del Barrio, and G. Botella, “Deep PeNSieve: A deep learning framework based on the posit number system”, *Digital Signal Processing: A Review Journal*, vol. 102, p. 102 762, 2020. DOI: 10.1016/j.dsp.2020.102762.

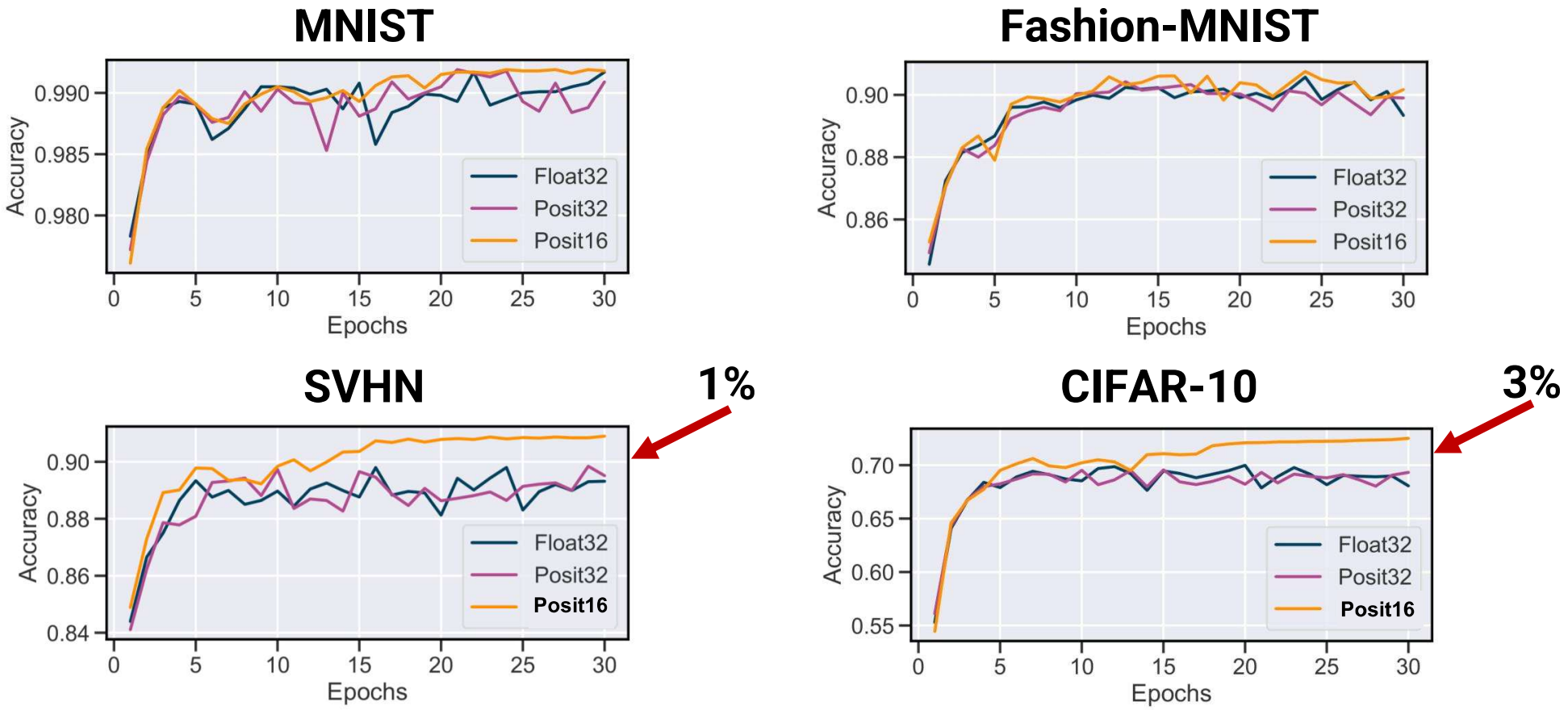
Deep PeNSieve

- A framework for posit-based DNNs
- Training stage: fully in the posit format
  - Posit⟨32,2⟩
  - Posit⟨16,1⟩
- Inference:
  - Posit⟨8,0⟩
  - Fused ops. (quire)



github.com/RaulMurillo/deep-pensieve

DNN training results



DNN post-training quantization

Format	MNIST		Fashion-MNIST		SVHN		CIFAR-10	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
Float16	99.17%	100%	89.34%	99.78%	89.32%	98.35%	68.05%	96.15%
INT8	99.16%	100%	89.51%	99.79%	89.33%	98.38%	68.15%	96.14%
Posit⟨8,0⟩	98.77%	99.99%	88.52%	99.82%	81.31%	97.07%	43.89%	86.49%
Posit⟨8,0⟩ with quire	**99.07%**	**99.99%**	**89.92%**	**99.81%**	**89.13%**	**98.39%**	**68.88%**	**96.47%**

Conclusions

Conclusions

- Designed parameterized Posit algorithms
  - Addition, multiplication & MAC
  - Integrated into the FloPoCo framework
- Developed Deep PeNSieve, an open-source posit-based DNNs framework
  - All computations performed in posit format
  - Promising results for posits in both training (16 bits) and inference (8 bits)
- Posits are still in development – there is room for improvement

Thank you!

Posit Arithmetic Units for Deep Neural Networks

Jornadas SARTECO 2021
Workshop CABAHLA

Raul Murillo, Alberto A. del Barrio & Guillermo Botella