Design and Implementation of a Quantized CNN Inference Engine on FPGA

INDRAHAS REDDY VENNAPUSA
19D070067

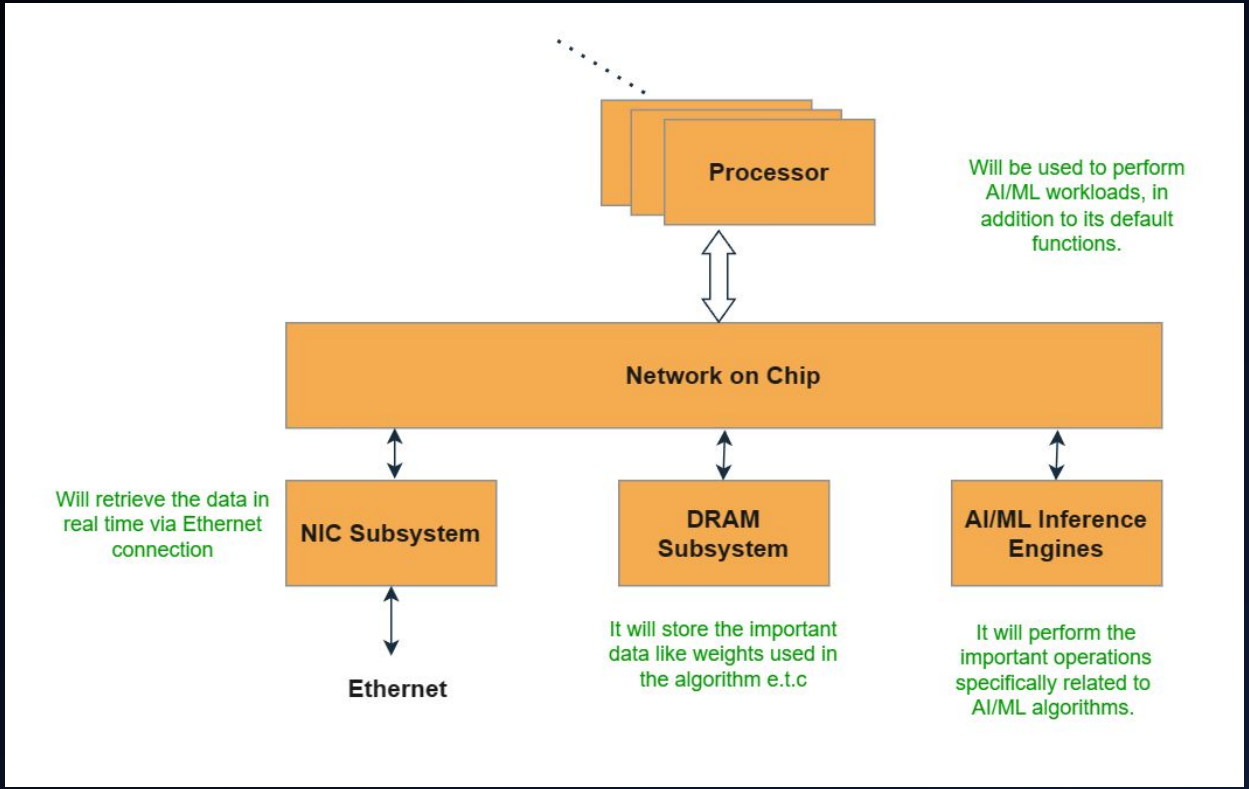
SUPERVISOR : PROF.MADHAV DESAI
DEPARTMENT OF ELECTRICAL ENGINEERING
IIT BOMBAY

Outline

- Introduction
- AI/ML Inference Engine
- SBC Architecture
- Inference Engine Architecture
- Results

Introduction

- The Big Picture – AI/ML accelerator at the edge.



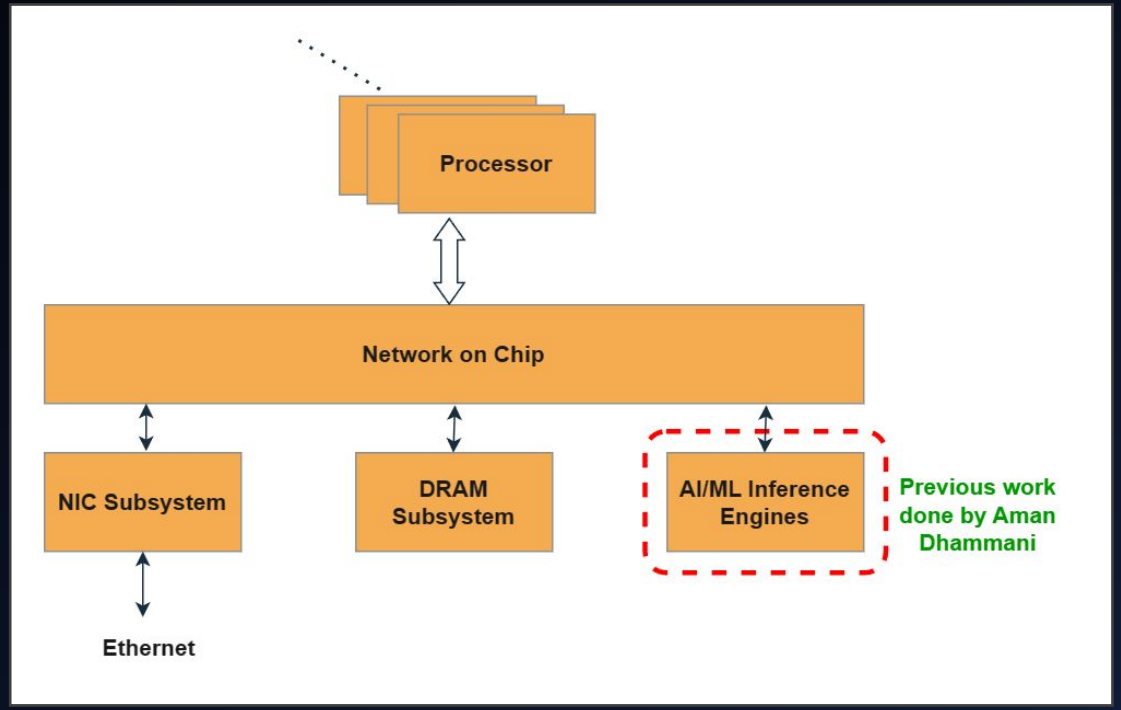
Introduction

- Why are AI/ML accelerators important?
  - AI and ML workloads are becoming increasingly complex and computationally demanding.
- Benefits of using AI/ML accelerators at the edge
  - **Smart security cameras**: AI/ML accelerators are used to power smart security cameras that can detect and identify people, objects, and events in real time.
  - **Self-driving cars**: AI/ML accelerators are used to power the complex algorithms that enable self-driving cars to navigate the road and avoid collisions
  - **Retail self-checkout kiosks**: AI/ML accelerators are used to power retail self-checkout kiosks that can recognize and scan items quickly and accurately.
    E.g Amazon Go

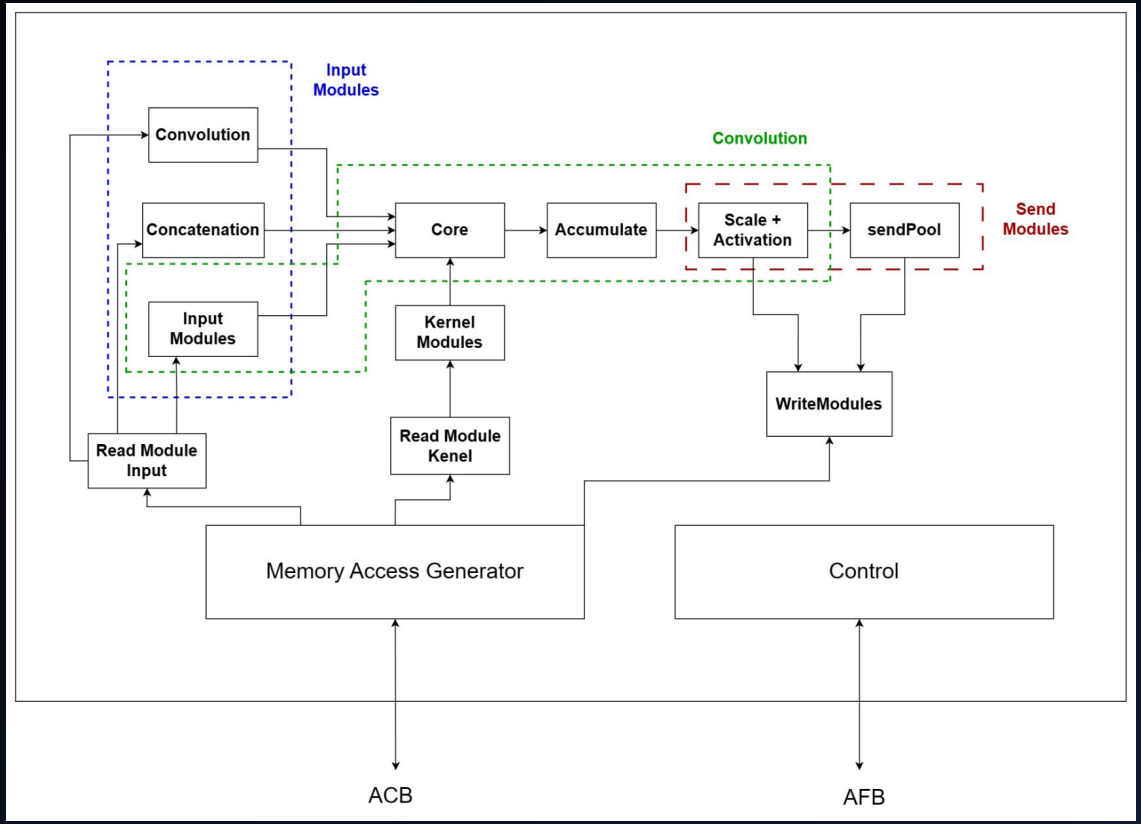
Introduction

- Types of AI/ML accelerators
  - GPUs (graphics processing units) – Highly complex and high power consumption
  - CPUs (central processing units) - Lower performance
  - FPGAs (field-programmable gate arrays)
  - ASIC (Application Specific Integrated Circuits) – High development costs and long development time

AI/ML INFERENCE ENGINE

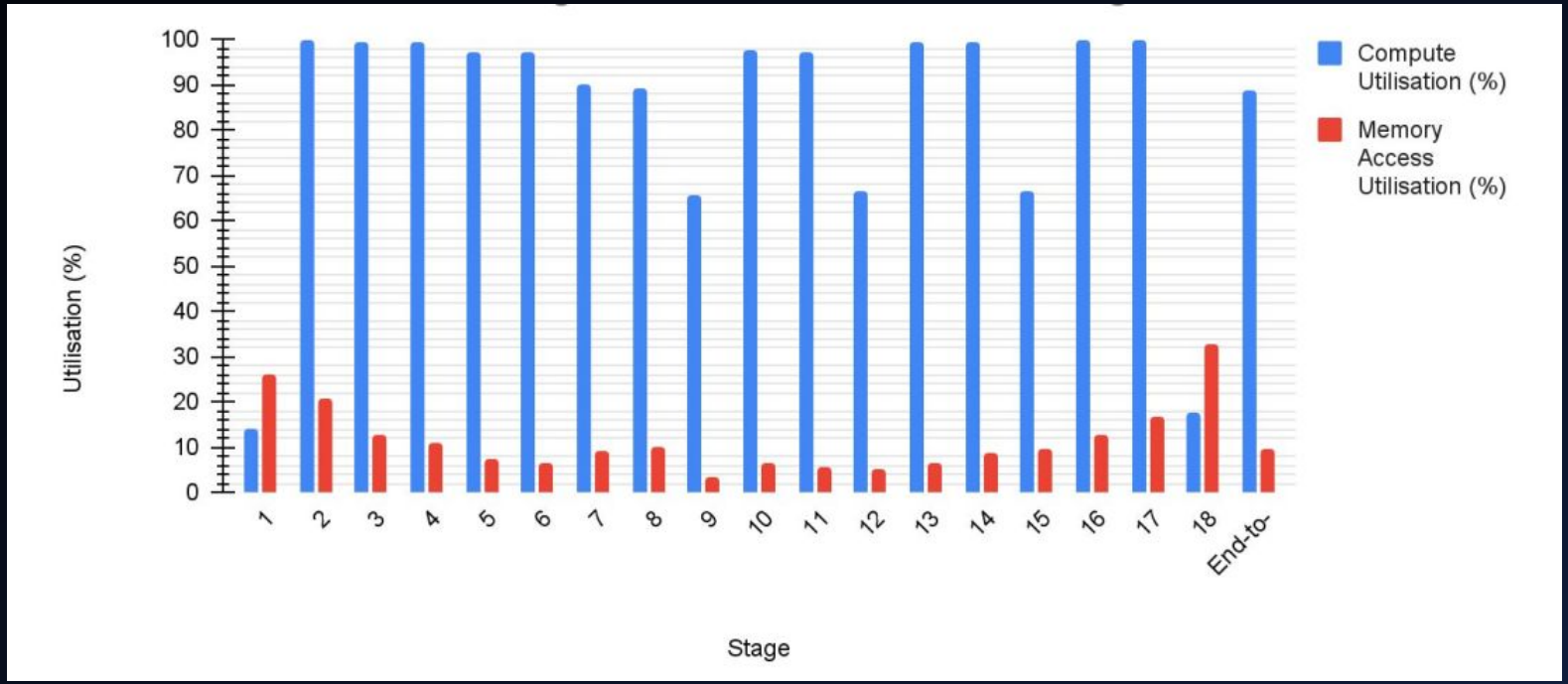


AI/ML Inference Engine - by Aman Dhammani



AI/ML Inference Engine - by Aman Dhammani

- Utilization of Accelerator Engine Resources



LeNet - 5 Architecture

- Dataset - MNIST Handwritten Digit
- A convolution neural network(CNN) architecture designed for image recognition tasks, comprising two convolution layers and three fully connected layers

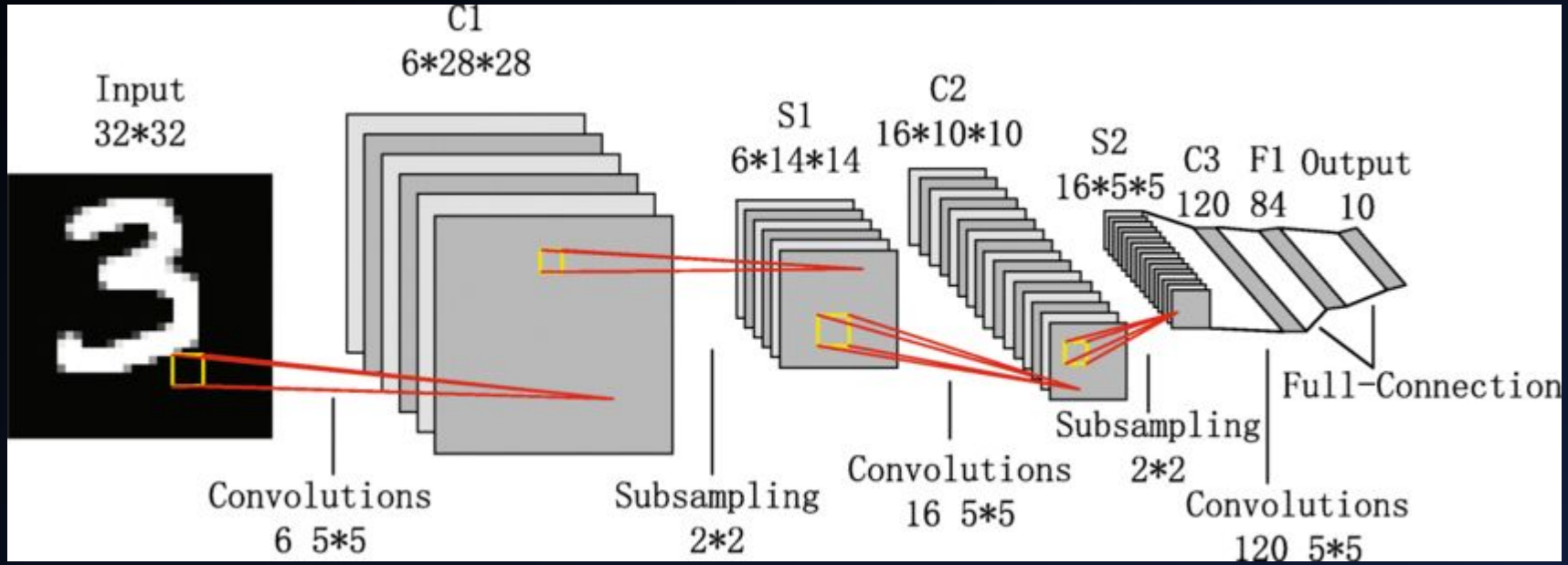
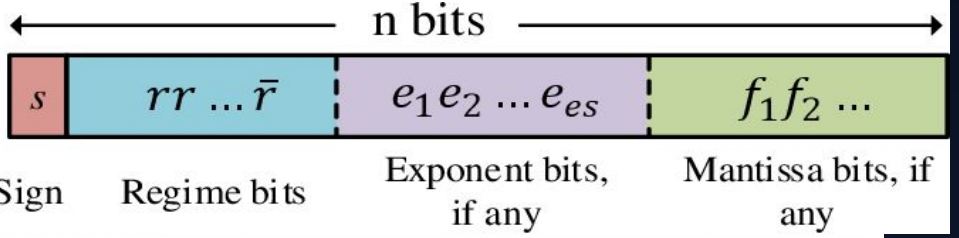


Image Source:(PDF) JPEG steganalysis based on ResNeXt with Gauss partial derivative filters (researchgate.net)

Posit Datatype- By Jitendra

- Dynamic range
- Alternative to floating-point numbers
- More efficient and compact representation

$$x = \begin{cases} 0, & p = 0, \\ \pm\infty, & p = -2^{n-1}, \\ sign(p) \times useed^k \times 2^e \times f, & \text{all other } p. \end{cases}$$



Size, Bits	IEEE Float Exp. Size	Approx. IEEE Float Dynamic Range	Posit *es* Value	Approx. Posit Dynamic Range
16	5	6×10^{-8} to 7×10^{4}	1	4×10^{-9} to 3×10^{8}
32	8	1×10^{-45} to 3×10^{38}	3	6×10^{-73} to 2×10^{72}
64	11	5×10^{-324} to 2×10^{308}	4	2×10^{-299} to 4×10^{298}
128	15	6×10^{-4966} to 1×10^{4932}	7	1×10^{-4855} to 1×10^{4855}
256	19	2×10^{-78984} to 2×10^{78913}	10	2×10^{-78297} to 5×10^{78296}

Image Source:unum (posithub.org)

Posit Datatype

Sr. No	Datatype for computation	Datatype for storage	Training Accuracy	Testing Accuracy
1	32 bit float	32 bit float	98.6%	98.27%
2	16 bit float	16 bit float	97.93%	97.23%
3	32 bit float	8 bit posit	88%	89%
4	16 bit float	8 bit posit	88.7%	89.37%

Post Training Static Quantization

- Technique for optimizing and compressing neural networks
- Converts floating-point numbers to integers for efficient inference
- Preserves model accuracy while reducing memory usage and compute requirements

$$Q(x, \text{scale}, \text{zero\_point}) = \text{round}\left(\frac{x}{\text{scale}} + \text{zero\_point}\right)$$

QUANTIZATION

$$unquantized\_value = (quantized\_value - zero\_point) \times scale$$

DEQUANTIZATION

Image Source:Introducing Post-Training Model Quantization Feature and Mechanics Explained (datature.io)

Post Training Static Quantization

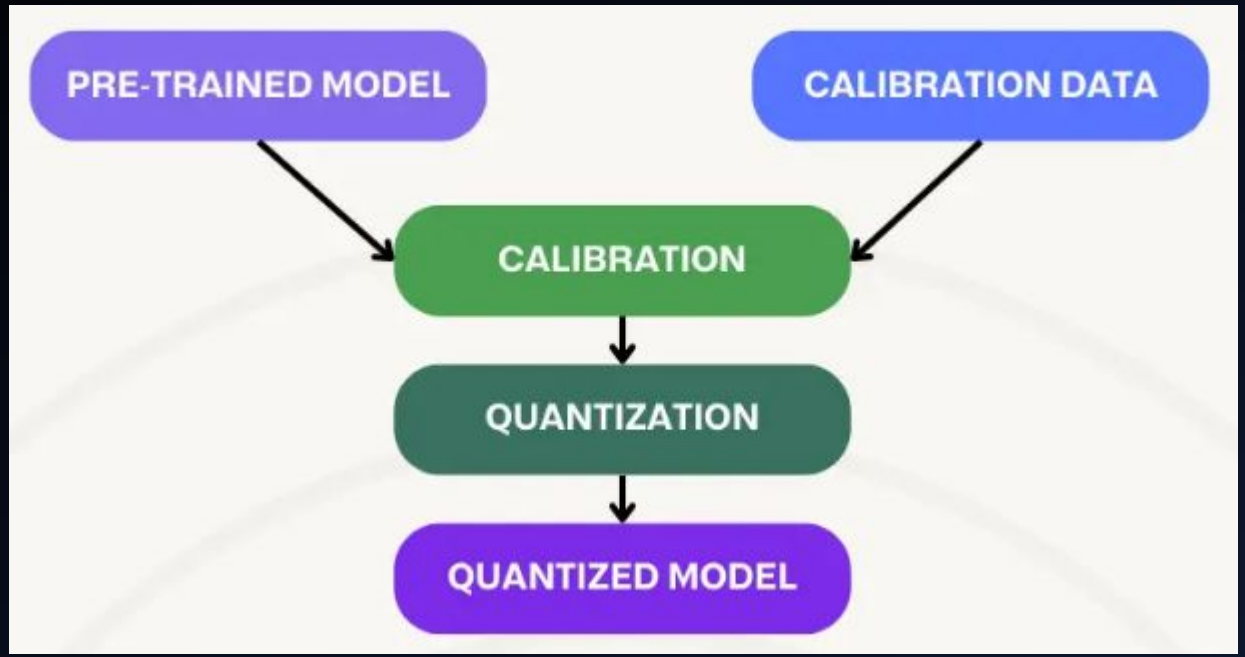
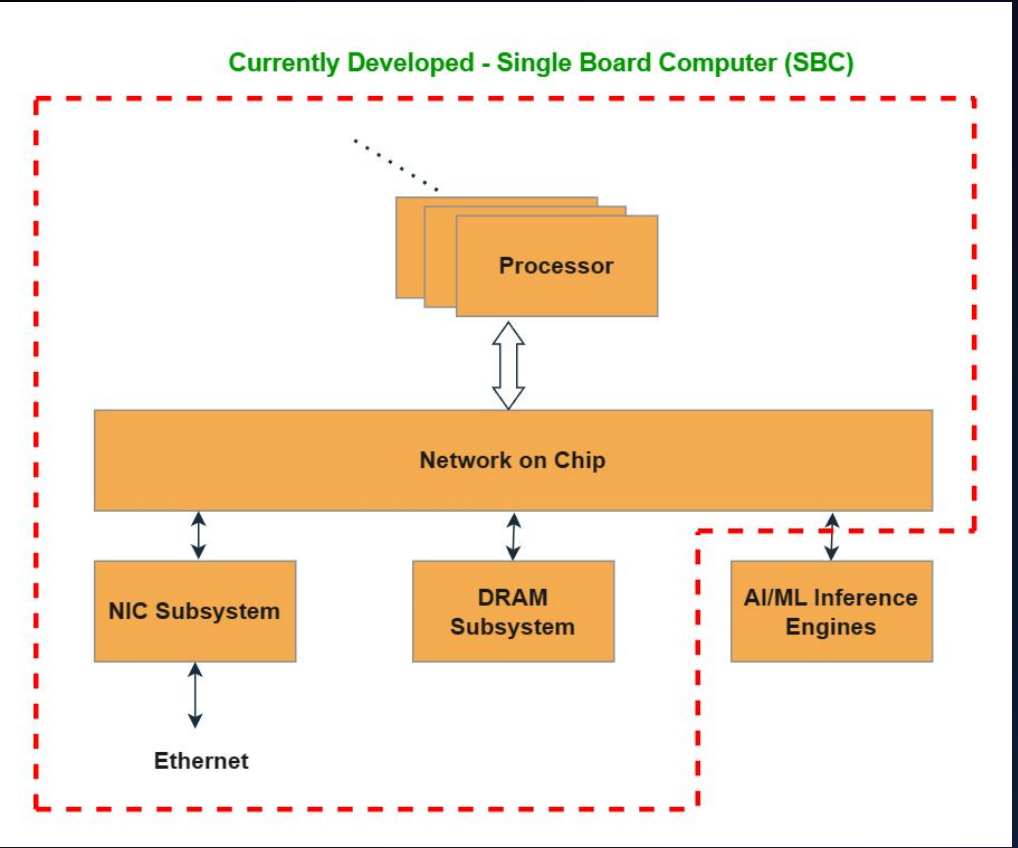


Image Source:Introducing Post-Training Model Quantization Feature and Mechanics Explained (datature.io)

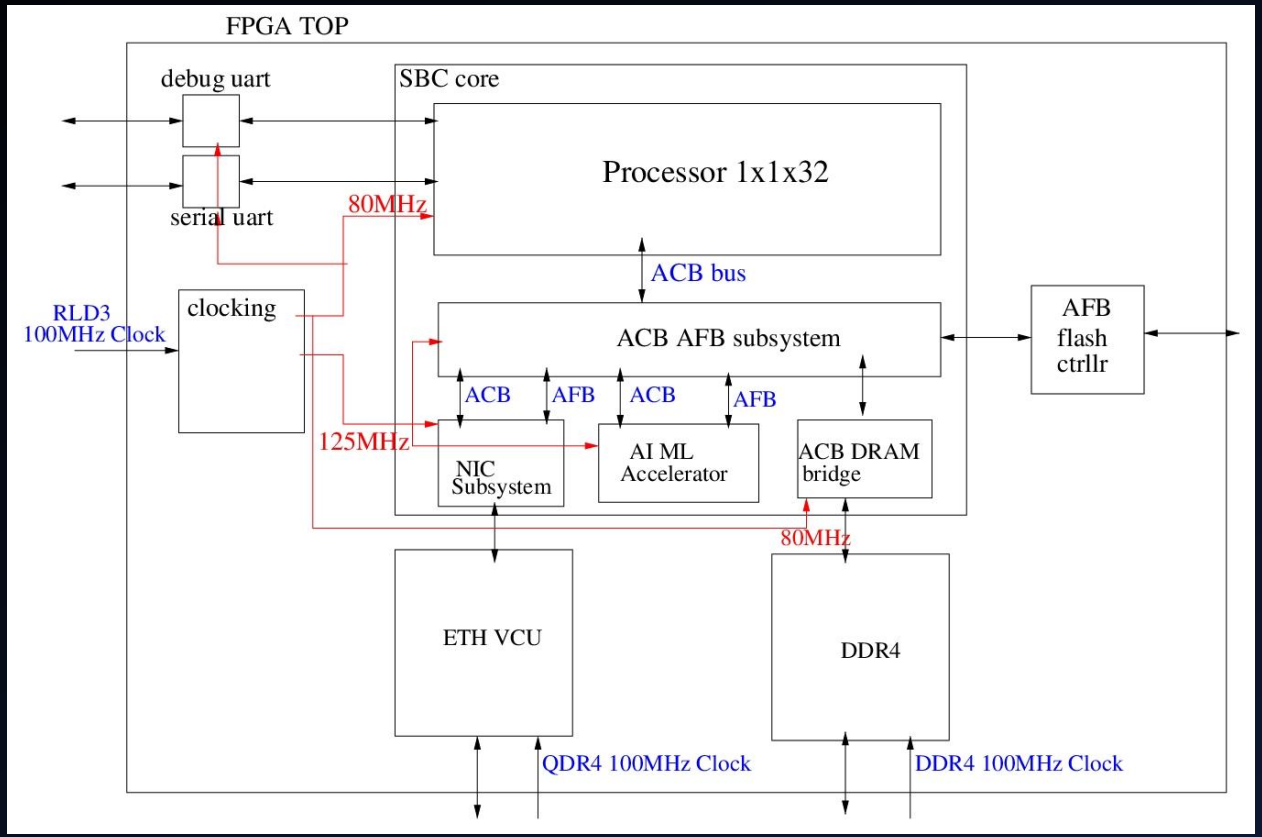
Post Training Static Quantization

Sr. No	Datatype for computation	Datatype for storage	Training Accuracy	Testing Accuracy
1	32 bit float	32 bit float	98.6%	98.27%
2	32 bit float	8 bit UINT	-	98.01%

SINGLE BOARD COMPUTER

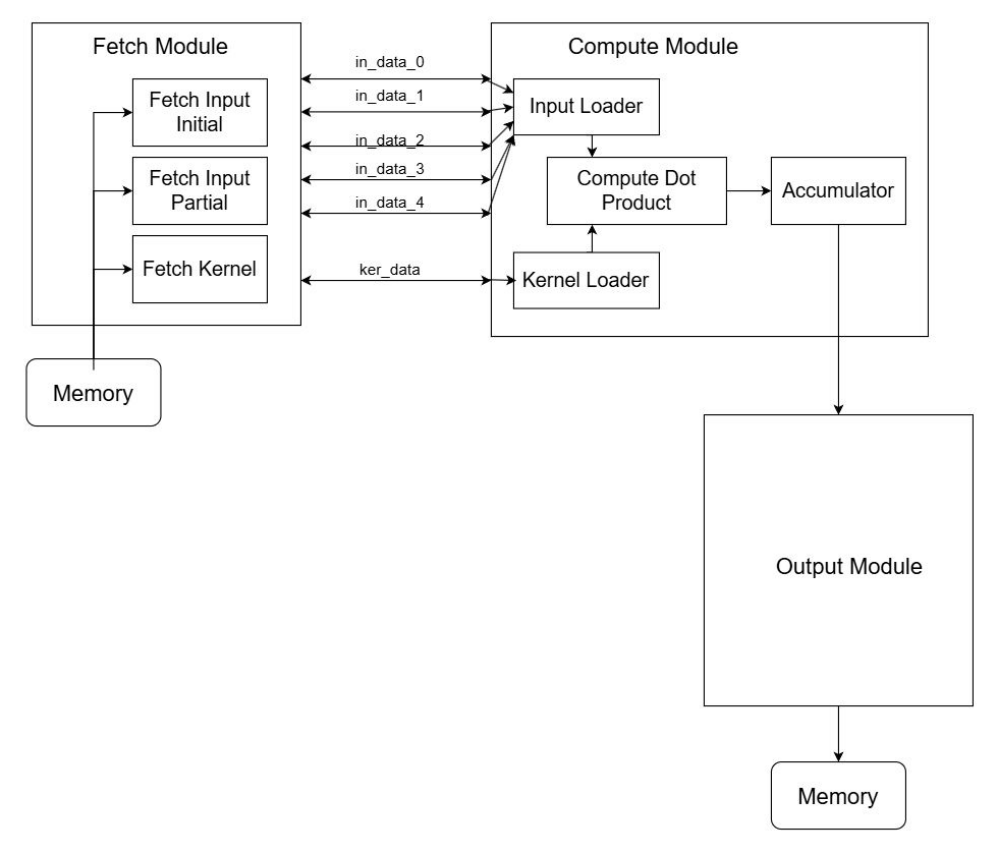


SBC Architecture



Inference Engine Architecture

- Designed using PyAHIR tool, developed by Rutuja, which generates AA code from Python



Convolution operation

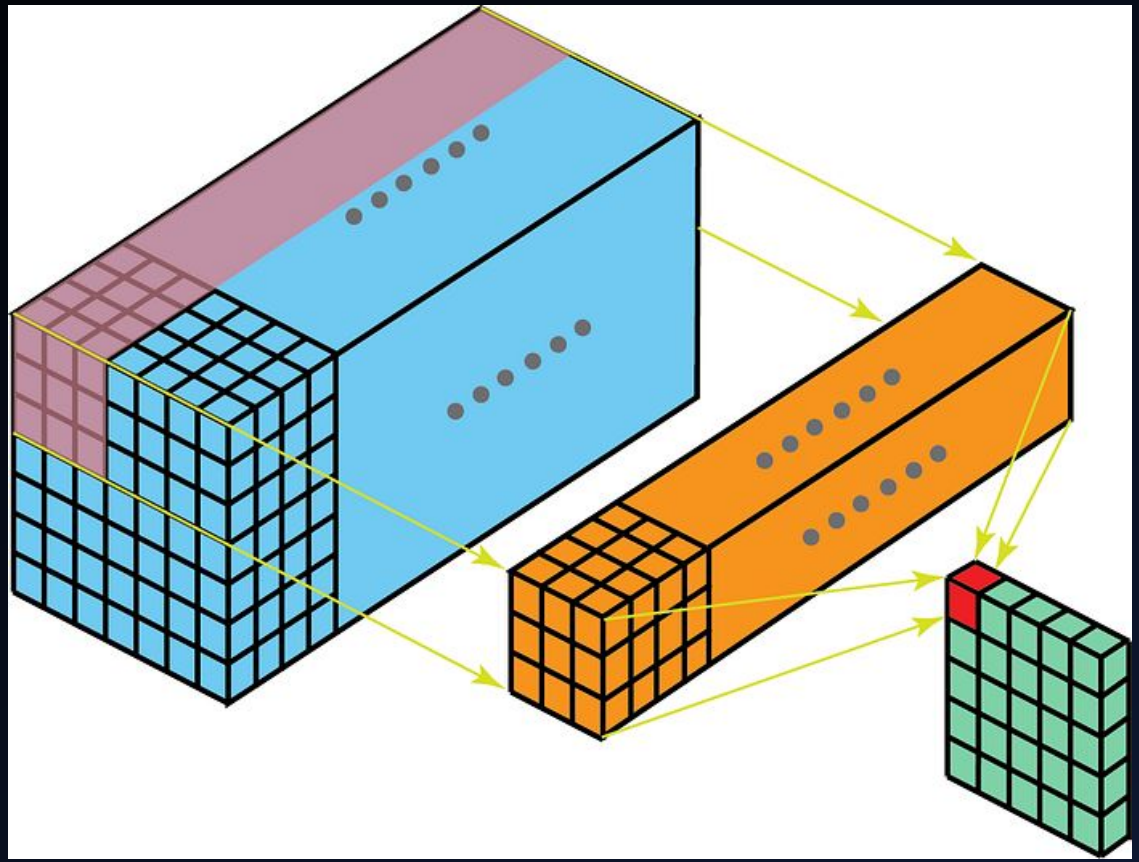
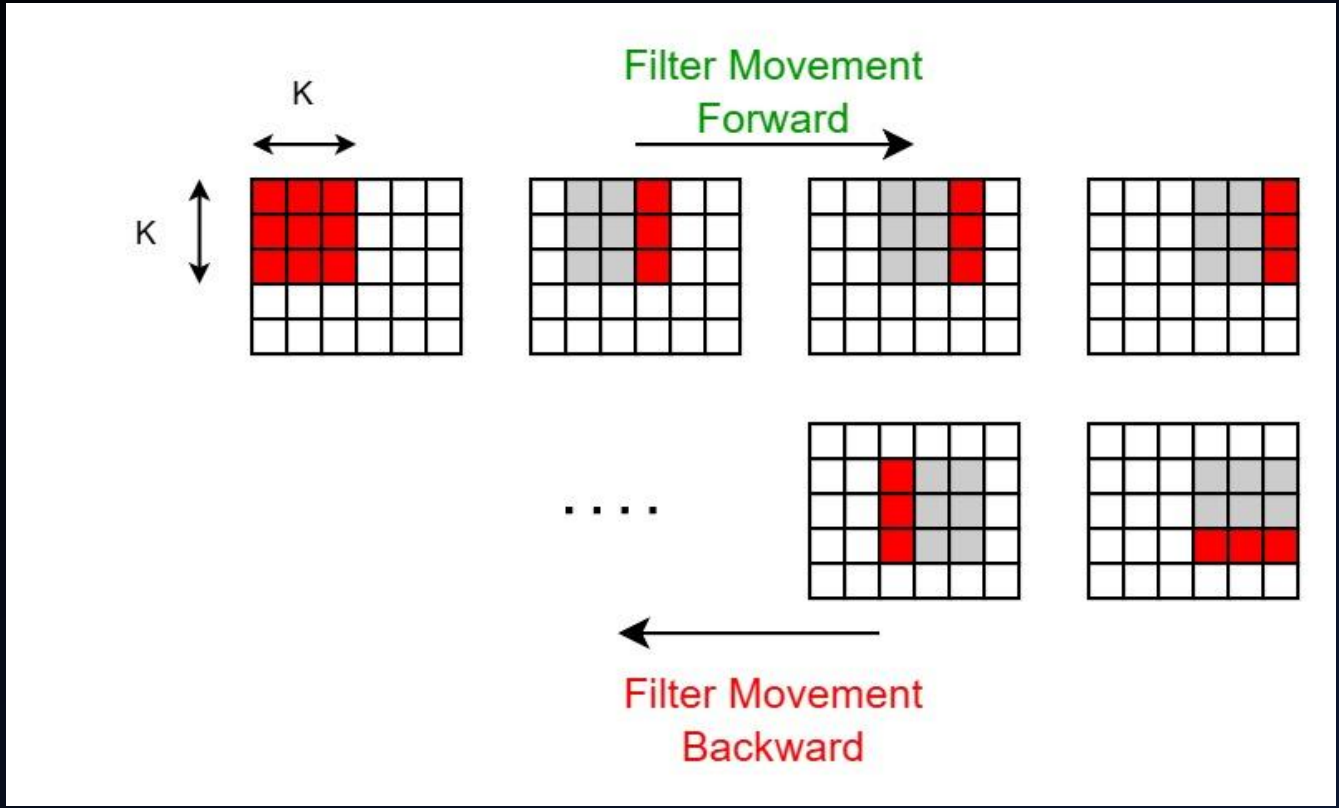


Image Source:convolutional neural networks - When should I use 3D convolutions? - Artificial Intelligence Stack Exchange

Convolution operation



Hardware Testing and Results

- Total inference time per image is 60 ms at 125MHz clock

```
In [62]: accuracy = (num_correct.item()/num_samples)*100
         print("Model Predicted {} correctly out of {} from testing dataset, Acuracy : {:.2f}".format(num_correct.item(),

Model Predicted 9800 correctly out of 10000 from testing dataset, Acuracy : 98.00
```

Platform	Datatype for Storage	Accuracy
CPU	Float 32	98.27
CPU	UINT8	98%
FPGA	UINT8	98%

Resource Utilization

Reference	Quantization Method	Clock	LUT	DSP	FF	Accuracy
[2]Mohd, Bassam et.al	Quantization Aware Training (QAT)	200 MHz	50k	-	-	96.67%
[3] Blott, Michaela et.al	FINN	250 MHz	67k	0	-	98.8%
[4] Ji, Mengfei et.al	FPQNet	250 MHz	53k	2614	94k	98.8%
This Work	PTQ	125 MHz	67k	183	74k	98%

THANK YOU

References

1. Aman Dhammani, “An AI/ML Inference Engine Developed Using AHIR-V2 Tools”, [Unpublished Dual Degree Thesis, IIT Bombay].
2. Mohd, Bassam & Ahmad Yousef, Khalil & Almajali, Anas & Hayajneh, Thaier. (2024). Quantization-Based Optimization Algorithm for Hardware Implementation of Convolution Neural Networks. Electronics. 13. 1-25. 10.3390/electronics13091727.
3. Blott, Michaela & Preußer, Thomas & Fraser, Nicholas & Gambardella, Giulio & O'Brien, Kenneth & Umuroglu, Yaman & Leeser, Miriam & Vissers, Kees. (2018). FINN-R: An End-to-End Deep-Learning Framework for Fast Exploration of Quantized Neural Networks. ACM Transactions on Reconfigurable Technology and Systems. 11. 1-23. 10.1145/3242897.
4. Ji, Mengfei & Al-Ars, Zaid & Hofstee, H.P. & Chang, Yuchun & Zhang, Baolin. (2023). FPQNet: Fully Pipelined and Quantized CNN for Ultra-Low Latency Image Classification on FPGAs Using OpenCAPI. Electronics. 12. 4085. 10.3390/electronics12194085.

References

- Y. Ma, Y. Cao, S. Vrudhula, and J.-S. Seo, ‘‘Optimizing loop operation and dataflow in FPGA acceleration of deep convolutional neural networks,’’ in Proc. ACM/SIGDA Int. Symp. Field-Program. Gate Arrays, 2017, pp. 45–54.
- Machine Learning Accelerator in Looker expands access to ML | Google Cloud Blog
- AI Accelerator - AWS Trainium - AWS (amazon.com)
- Machine Learning Accelerator | IBM