# Flash Attention on FPGA

Aakarsh A

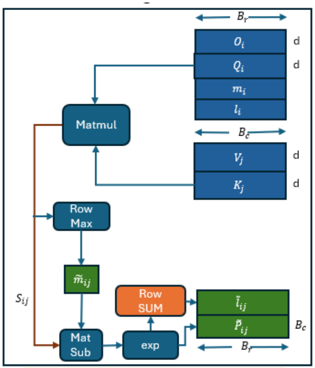
Department of Computer Science and Automation
Indian Institute of Science

May 30, 2025

# Overview

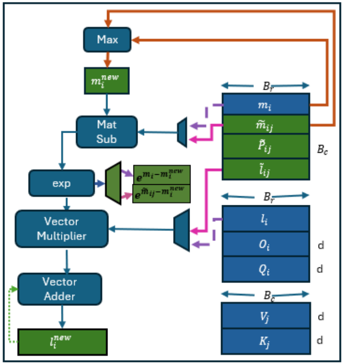
1. Flash Attention Implementation in brief
2. $e^{-x}$ Implementation
3. Future Work

# Block Level Design



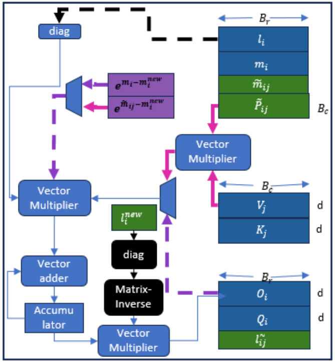
**Phase 1 : Calculation of $\tilde{m}_{ij}, \tilde{P}_{ij}$ and $\tilde{l}_{ij}$**

- $S_{ij} = Q_i K_j^T$
- $\tilde{m}_{ij} = rowmax(S_{ij})$
- $\tilde{P}_{ij} = \exp(S_{ij} - \tilde{m}_{ij})$
- $\tilde{l}_{ij} = rowsum(\tilde{P}_{ij})$



**Phase 2 : Calculation of $m_i^{new}$ and $l_i^{new}$**

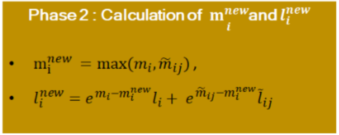
- $m_i^{new} = \max(m_i, \tilde{m}_{ij})$ ,
- $l_i^{new} = e^{m_i - m_i^{new}} l_i + e^{\tilde{m}_{ij} - m_i^{new}} \tilde{l}_{ij}$



**Phase 3 : Calculation of $O_i$**

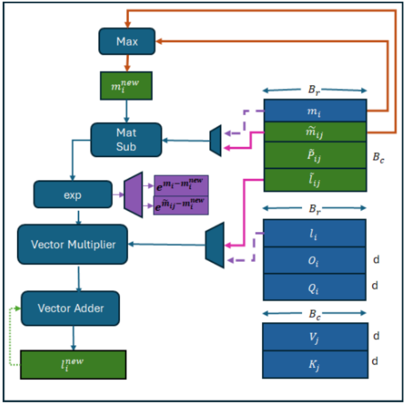
- $O_i = diag(l_i^{new})^{-1} \times \left( diag(l_i) e^{m_i - m_i^{new}} O_i^{old} + e^{\tilde{m}_{ij} - m_i^{new}} P_{ij}^{\sim} V_j \right)$

# Focus on Phase - 2



Phase 2 : Calculation of $m_i^{new}$ and $l_i^{new}$

- $m_i^{new} = \max(m_i, \tilde{m}_{ij})$,
- $l_i^{new} = e^{m_i - m_i^{new}} l_i + e^{\tilde{m}_{ij} - m_i^{new}} \tilde{l}_{ij}$



## RTL Components

- Negative Exponential Engine
- BRAM/URAM Access and Storage
- Timing and Control Unit
- Vector Multiplier
- Vector Adder
- Subtractor
- Comparator

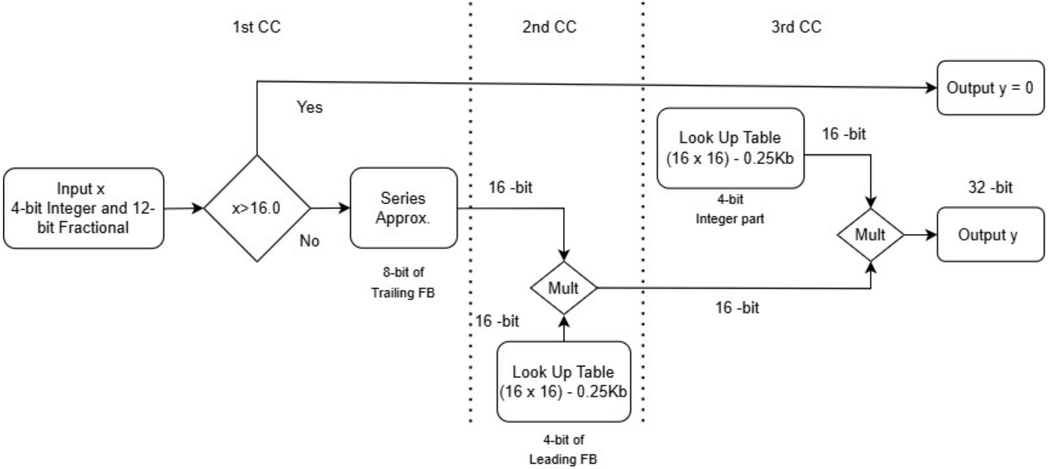
# Why Hybrid Approach?

Our goal: Hardware acceleration.

- Faster Memory Access (Infer LUTs) at the expense of hardware (DSPs).
- Minimize Latency and maximize operation frequency.

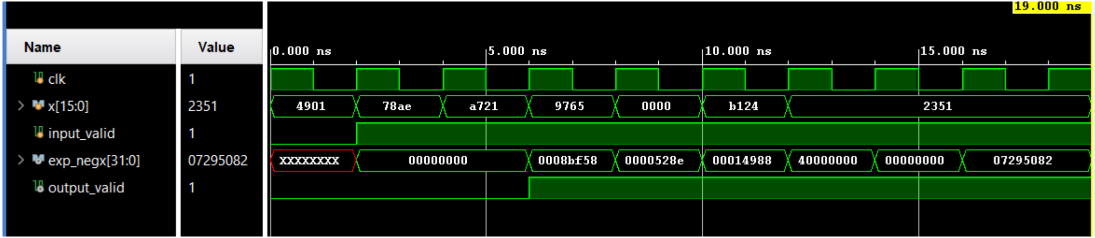
# Hybrid Approach

## Pipelined Dual LUT & Series Approximation



Latency is 2 clock cycles and Throughput is 1. DSPs used: 2

# Simulation Results



- Output is **32** bits: 2 Integer Bits and 30 Fractional bits
- Achieved a maximum error of less than 0.0005 (0.05 % error in values).
- The previous implementation using $2^{\frac{-x}{\ln 2}}$ was replaced due to higher latency when pipelined and their difficulty for realizing fractions.

# Future Work

1. Accuracy comparison between different implementation techniques.
   Target error: Less than 0.005
2. RTL implementation of the Timing and Control Unit, along with other subunits,
3. Verification of Phase-2 on the Alveo V80 FPGA.

# The End