EE23M053 Prabhav Talukdar

EE23M100 Tonda Pavankumar

CS6230: CAD for VLSI Systems

Project 1:

Multiply-Accumulate (MAC) Unit Design

Implement and verify the MAC module using Bluespec System Verilog (BSV), as mentioned in the specification given. Students are requested to maintain a github repository where the source code and a detailed report, with description of microarchitecture and design/verification methodologies used, are uploaded.

Description:

MAC operation: A*B + C

Designing a Multiply-Accumulate (MAC) module that supports the MAC operations for the following data types.

- S1. (A: int8 , B: int8 , C: int32)-> (MAC: int32)
- S2. (A: bf16, B: bf16, C: fp32)-> (MAC: fp32)

MAC Unit:

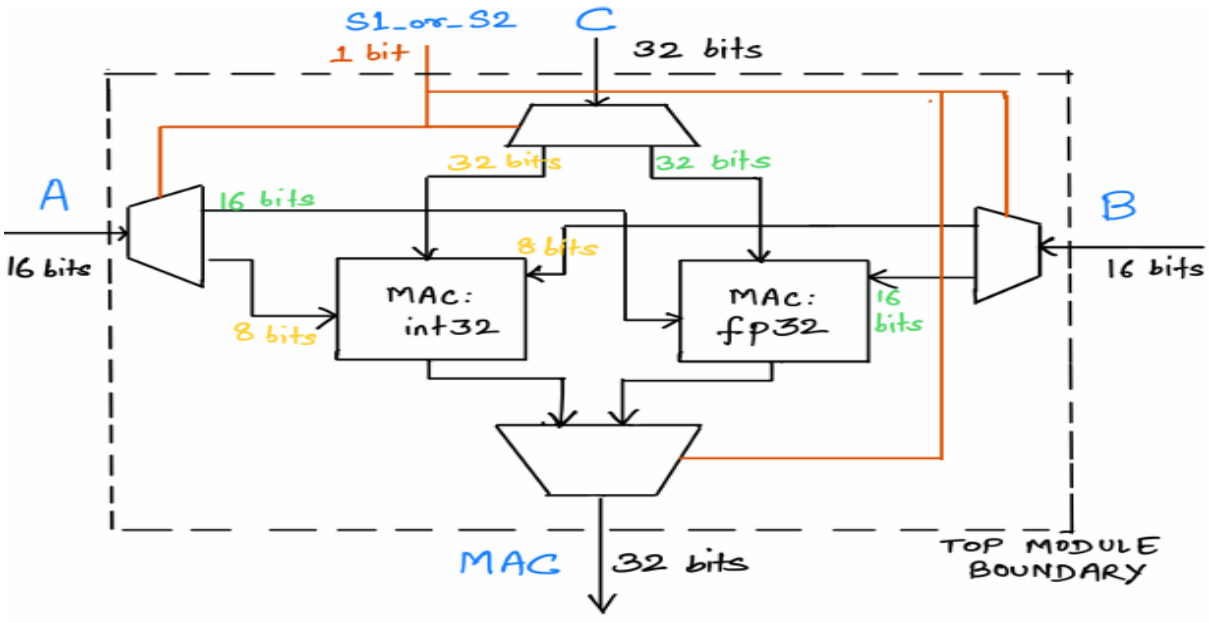


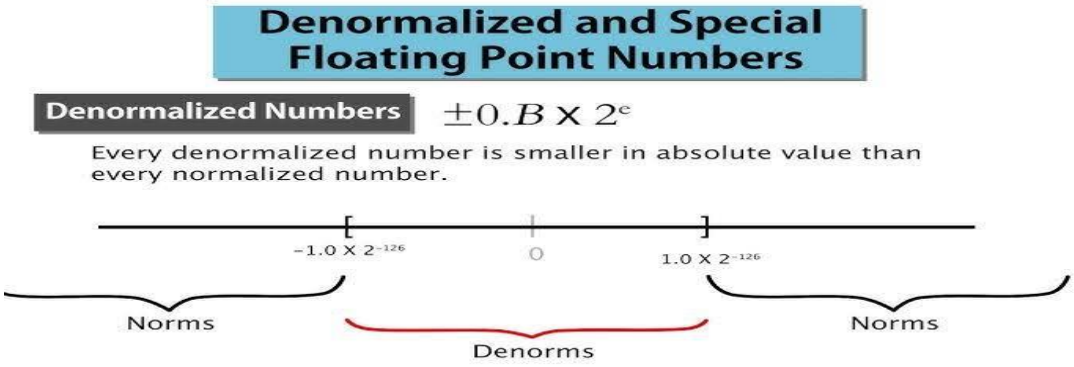
Figure 1. Architectural specification of the MAC unit

S1. (A: int8 , B: int8 , C: int32)-> (MAC: int32)

- For int8 multiplier, all combinations (256x256) have been tested.
- For adder, numbers close to positive infinity, negative infinity, zero have been tested.
- Over flow and underflow has been checked.

S2. (A: bf16, B: bf16, C: fp32)-> (MAC: fp32)

- Test cases have been generated from Kuda library.
- All the generated test cases are verified in Bluespec.



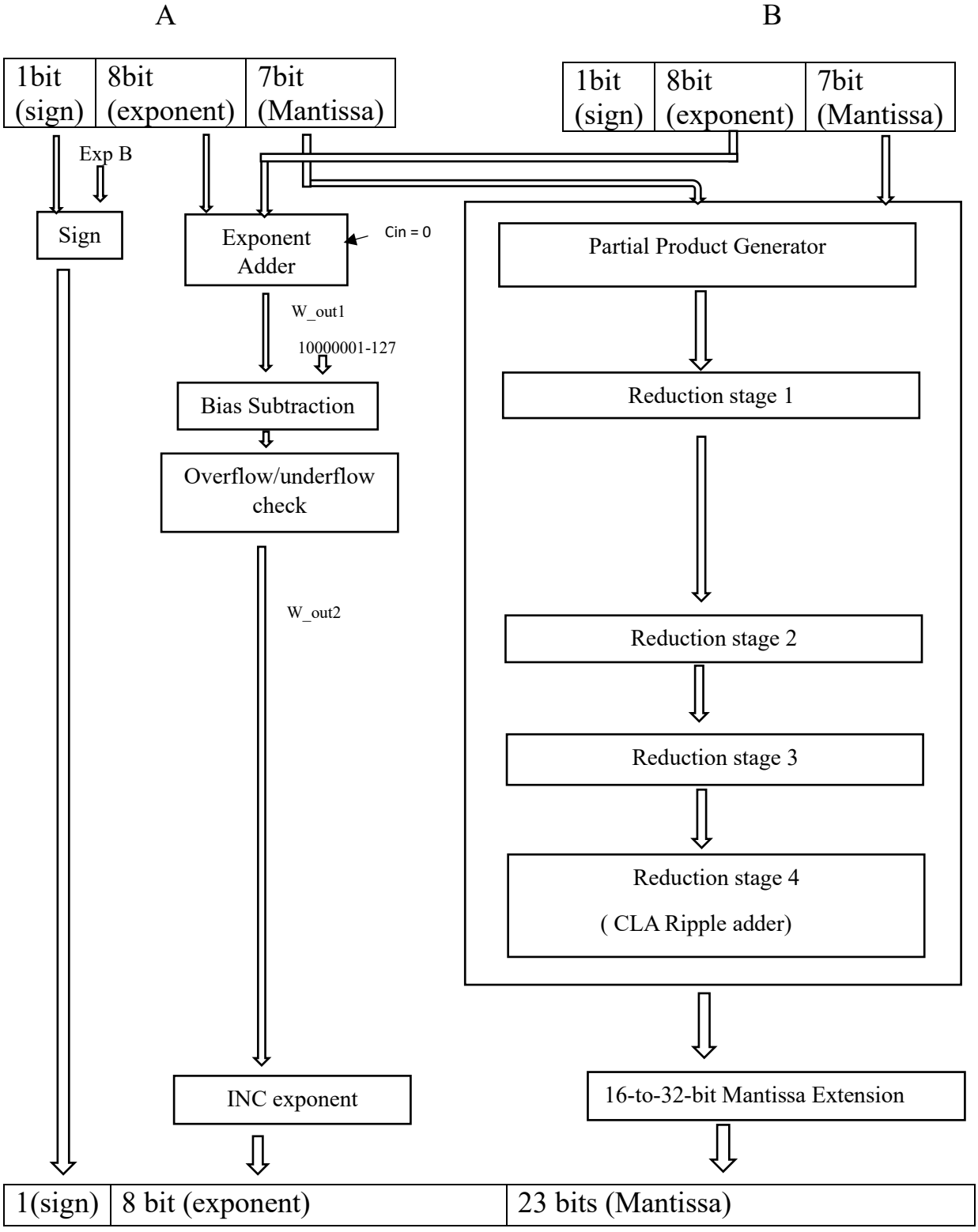
- Test values greater than 1.0 x 2 ^-126 and less than -1.0x2^-126 have been tested.
- Overflow and underflow have been checked.
- All the exceptions have been taken care of.

Unpipelined Design:

MAC Architecture MULTIPLIER

Inputs: A, B ----------→ bfloat16 numbers

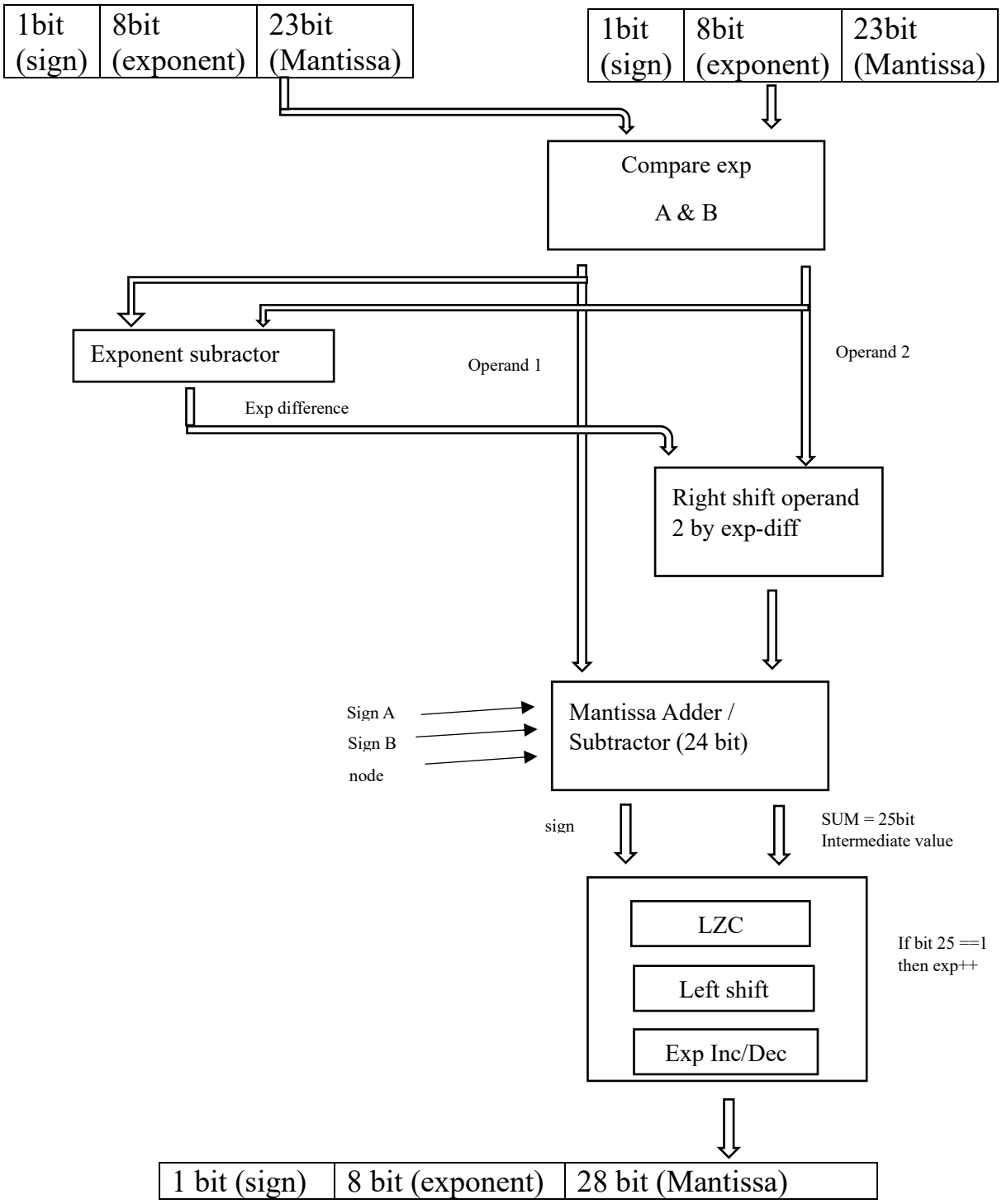
Output: A*B----------→ fp32 number



FP 32 Adder:

Inputs: A, B converted to FP32 and

Output: A*B+C

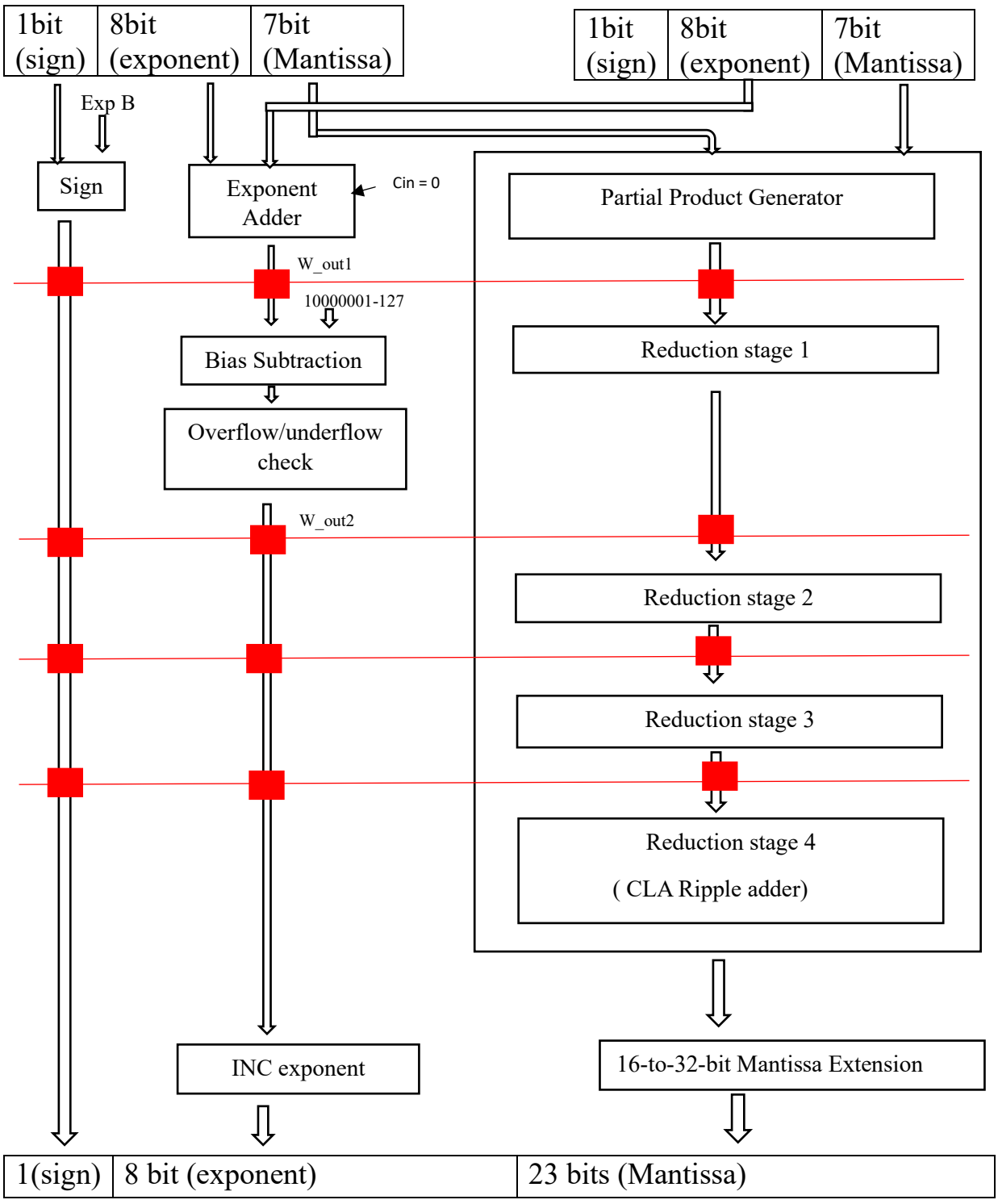


Pipelined Design:

MAC Architecture MULTIPLIER

Inputs: A, B ----------→ bfloat16 numbers

Output: A*B----------→ fp32 number



FP 32 Adder:

Inputs: A, B converted to FP32 and

Output: A*B+C