Amar Kumar-CED17I029

Pipelining of Half Precision Floating Point multiplier

Introduction

Floating point multiplication has been very important in the field of signal processing, graphics acceleration, image processing and many more.
Its complexity is high as it requires more area and hence consumes more power as compared to fixed point multipliers.

Half Precision Floating Point Representation

The IEEE half precision floating point standard representation requires a 16 bit word, which may be represented from 0 to 15, left to right. The first bit is the sign bit, S, the next five bits are the exponent bits, 'E', and the final 10 bits are the fraction 'F' i.e. also known as mantissa:

```
S  EEEEE  FFFFFFFFFF
0  1      5  6          15
```

Floating point Algorithm

1. Add exponent
2. Multiply mantissa
3. Normalize to get final mantissa and exponent
4. XOR sign bit to get sign

Pipelining, Logic components and comparison

(assume delay of the gates as following:- NOT-1D, AND-1D, OR-1D, XOR-2D)

Adder and subtractor used is carry save adder(CSA)
Wallace is 16bit with stages that reduce from 16 rows -> 11 rows -> 8 rows -> 6 rows -> 4 rows -> 3 rows -> 2 rows -> 1 row.
It uses full adder and each full adder has a delay of 2 XOR gates and 1 XOR gate delay is 2D. So total delay for wallace is 7*2*2 = 28D

Exponent sum is CSA which handles 5-8 bit (sign extended) inputs
The 8bit CSA this uses about 16 units for addition/subtraction
Normalizer detects carry from wallace to add to exponent and reduces 32bits->20bits and it has a delay of two gates.

The pipeline buffer storage delay is also 2D

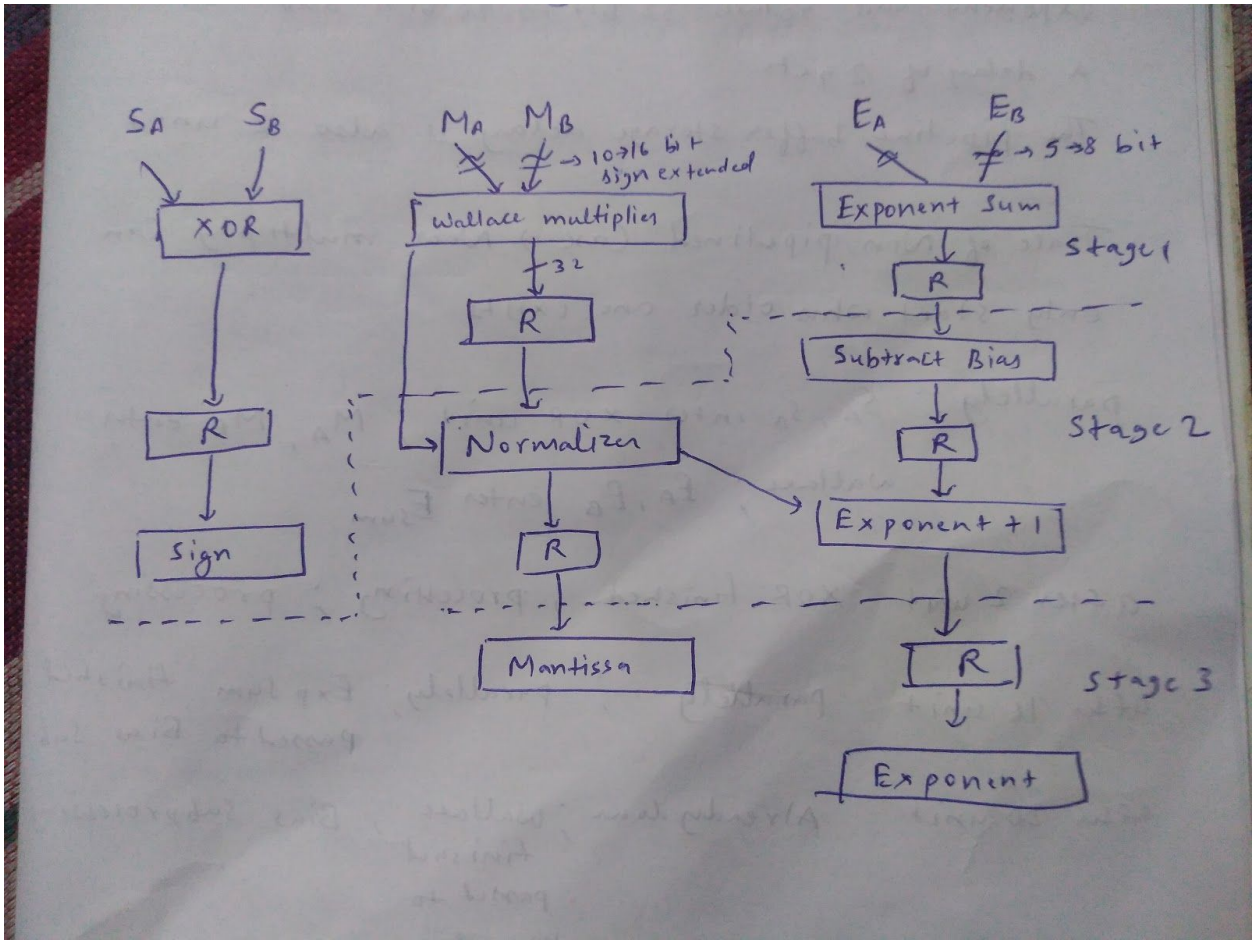
Trace of Non-pipelined case
New multiply can only start when older one exists.
Parallely: Sa,Sb enter XOR unit ; Ma,Mb enter wallace ; Ea,Eb enter Esum
After 2D - XOR finished ; processing ; processing
After 16D - parallely ; parallely ; Esum finished,passed to bias sub
After 20D - already done ; wallace finished,passed to normalize ; bias subprocessing
After 30D - already done ; normaliser done,passed to output ; processing bias subtraction
After 32D - already done ; output reached mantissa ; bias subtraction done,adding exponent+1(in case of overflow)
After 48D - already done ; output reached mantissa ; exponent reached

Till now multiplication of first instruction is done
For next multiplication another 48D is required

Hence 2 instructions has a delay of 96D

Pipelined case
Multiple instructions can be in different stages
Parallely: Sa,Sb enter XOR unit ; Ma,Mb enter wallace ; Ea,Eb enter Esum
After 2D - XOR finished ; processing ; processing
After 16D - XOR finished ; processing ; Esum done,pass to bias
After 20D - finished ; wallace done ; already done
Instruction 1 goes to stage 2
Instruction 2 goes to stage 1
After 30D - I2 finished ; I1 normaliser done,I2 in wallace(2D past) ; processing sub bias,Esum being evaluated
After 40D - I2 finished ; I1 finished,I2 under execution in wallace(20D passed) ; I1 finished, Esum finished 4 unit into sub bias
After 80D - I1 finished,I2 finished ; I1 finished,I2 finished ; I1 finished,I2 finished ;

In 80D , 2 instructions have been finished



% of Improvement can be achieved with respect to non-pipelined architecture.

%Improvement using pipeline = $\frac{non-pipelined}{pipelined} \; x \; 100$

%Improvement using pipeline = $\frac{96}{80} \; x \; 100 \; = \; 120\%$