

## Introduction

- Approximate computing has emerged as a potential solution for the design of energy-efficient digital systems. Applications such as multimedia, recognition and data mining are inherently error-tolerant and do not require a perfect accuracy in computation.
- A fast approximate adder that can process data in parallel is designed by cutting the carry propagation chain. A novel approximate multiplier design is proposed using a newly designed approximate adder.
- A simple tree of the approximate adders is used for partial product accumulation and the error signals are used to compensate the error for obtaining a better accuracy.
- Compared to the Wallace multiplier, a 16-bit approximate multiplier implemented in a 28nm CMOS process shows a reduction in delay and power of 20% and up to 69%, respectively.
- When applied in image processing applications, the approximate multiplier achieves similar performance as accurate multipliers with high PSNRs ( $\approx 50$  dB)

## Proposed Approximate Adder

- The proposed adder produces an error when the carry propagation chain (CPC) is cut; otherwise it is correct.
- Consider the case when a CPC is formed at the  $i$ th bit when  $A_{i-1} = B_{i-1} = 1$  and  $A_i \oplus B_i = 1$ . An accurate adder propagates the carry to higher bits, i.e.,  $C_i = C_{i+1} = 1$ , whereas the approximate adder only propagates the carry to its nearest neighbor, i.e.,  $C'_i = 1, C'_{i+1} = 0$ . Meanwhile, it generates an approximate sum  $S'_i = 1$  and an error signal  $E_i = 1$ .
- Equivalently, the carry at the  $(i+1)$ th bit is divided to an approximate sum and an error (which is used for error reduction), i.e.,  $C_{i+1} = S'_i + E_i$ . Thus the proposed adder cuts the CPCs that are longer than two bits.
- Error reduction is based on the following relationship ( $S$ : accurate sum,  $S'$ : approximate sum, and  $E$ : error).  
$$S = S' + E$$
- Summary: The carry signal only propagates one bit higher in any situation in this approximate adder. The proposed  $n$ -bit adder performs a bit-wise, parallel processing. It is even faster than a traditional full adder.

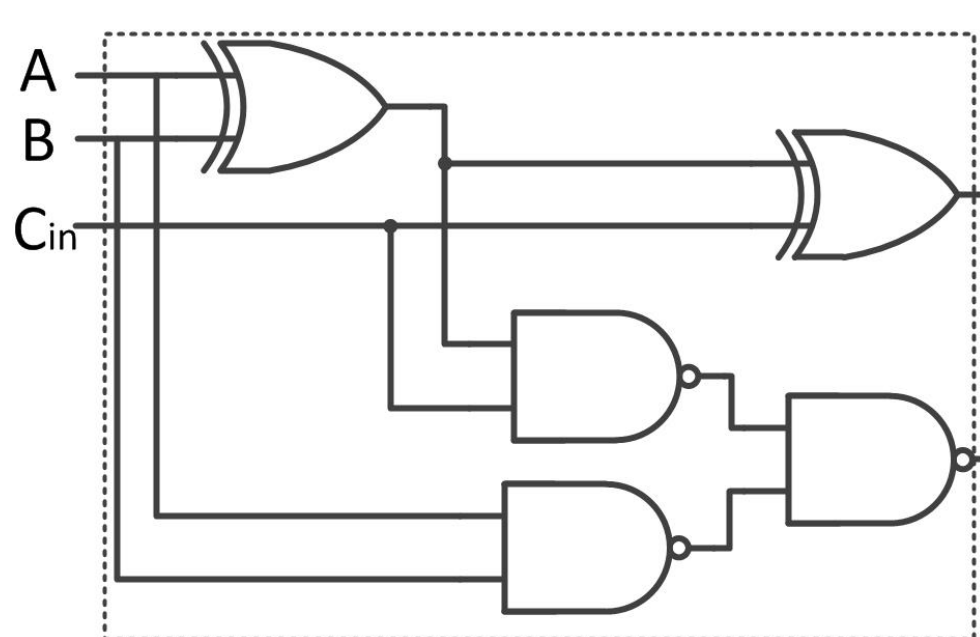


Fig. 1 A full adder

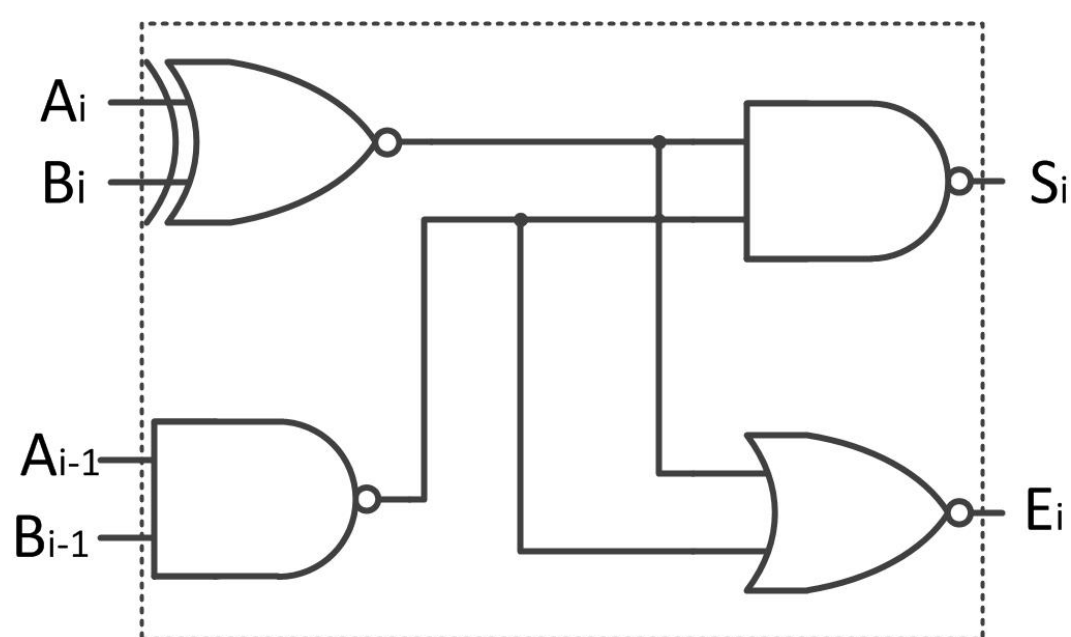


Fig. 2 An approximate adder

## Proposed Approximate Multiplier

- Partial product accumulation**  
Each approximate adder can accumulate two rows of partial products into one row.
- Error accumulation**
  - Errors can occur at each bit of the tree.
  - The errors are “sparse,” i.e., it's very rare to have multiple errors at the same position.
  - At each bit, the errors are accumulated by OR operations, which are approximate with an acceptable accuracy.
- Error reduction**
  - The accumulated error vector is added back to the result using an accurate adder.
  - Different number of MSBs of the error signals can be used according to different accuracy requirements.

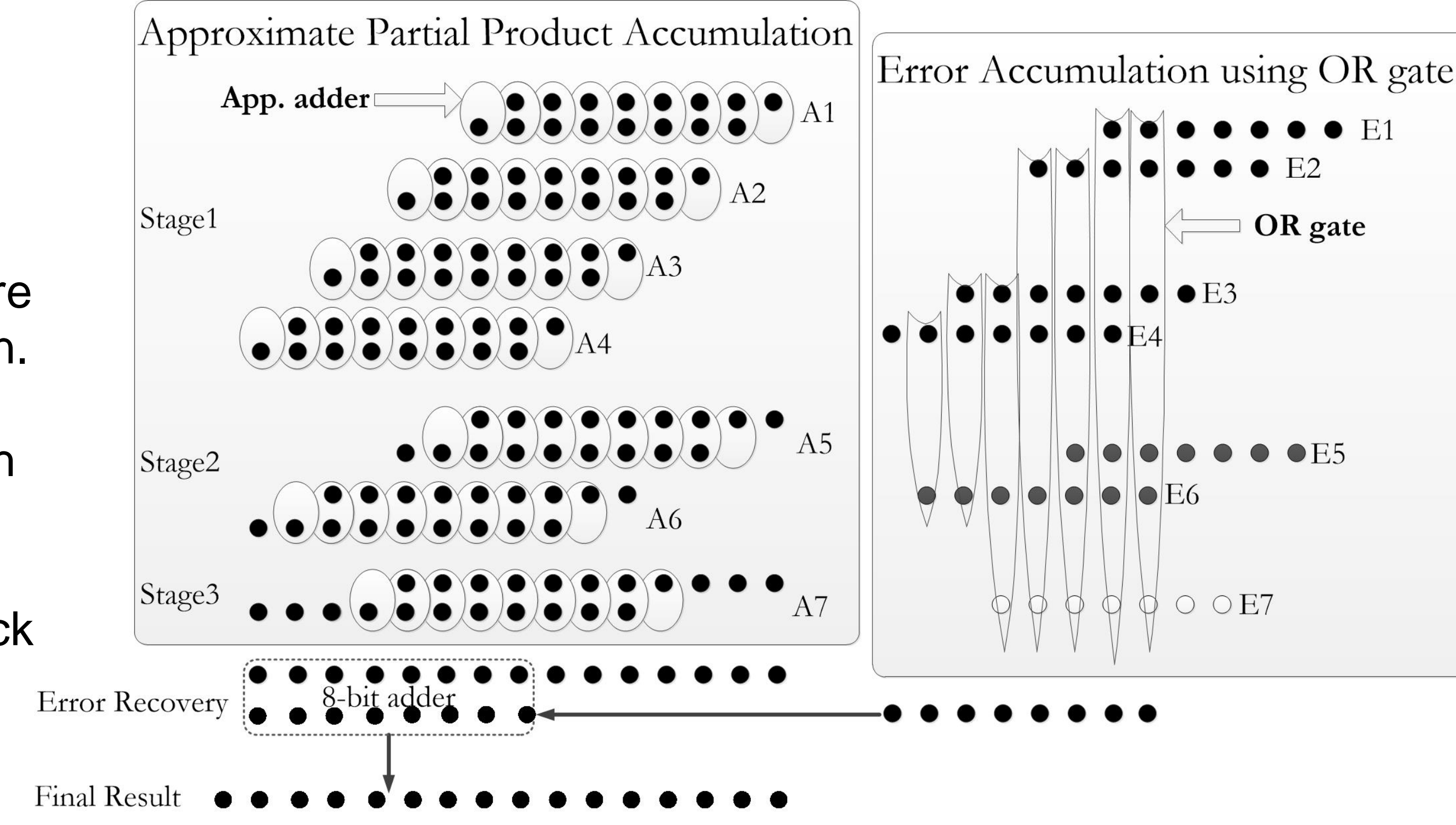


Fig. 3 The proposed multiplier

## Delay

- Delay Estimation**
  - Using a linear delay model, the delays of a full adder and an approximate adder are 3 and 2 "gate delays."
  - The delay of the partial product accumulation tree is estimated based on the linear model. There are approximately  $\log_2 n$  layers in the approximate adder and  $\log_{1.5} n$  layers in the Wallace tree.

$$D_{approx} = (2 \lceil \log_2 n \rceil + 1) \tau_g$$

$$D_{wallace} = 3 \lceil \log_{1.5} n \rceil \tau_g$$

Table 1. Delay comparison between the approximate and Wallace multipliers

Multiplier Size	8	16	32	64	$2^k$
Approximate	7	9	11	13	$2k + 1$
Wallace	12	18	24	30	$5k$

- Experimental Results**
  - $16 \times 16$  approximate and Wallace multipliers are implemented in both FPGA and STM 28nm process.
  - In the FPGA implementation, the proposed design reduces delay by 36.4%.
  - In the ASIC implementation, the proposed design reduces delay by 20%.

## Power Consumption

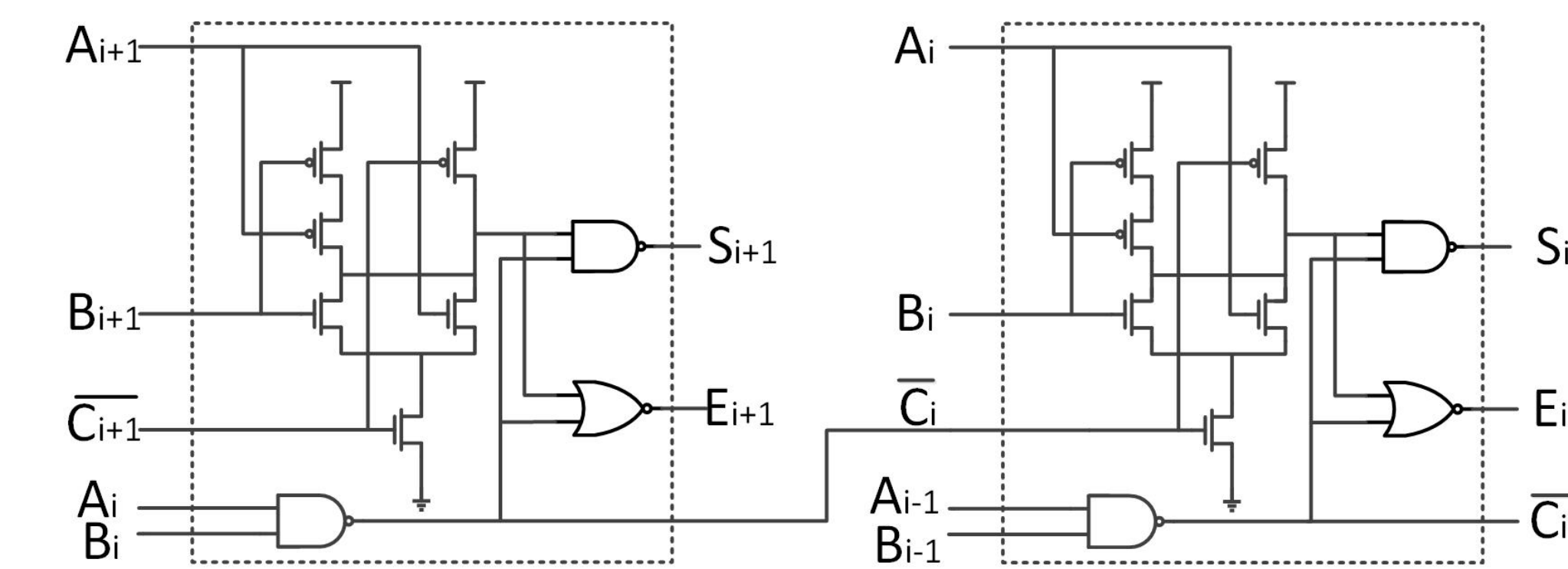


Fig. 4 ASIC Implementation of two approximate adder cells using shared logic

- In the FPGA implementation, the proposed design reduces dynamic power by 44.3%.
- In the ASIC implementation, the proposed design reduces power by 48-69% @ 0.1, 0.25 and 1 GHz.

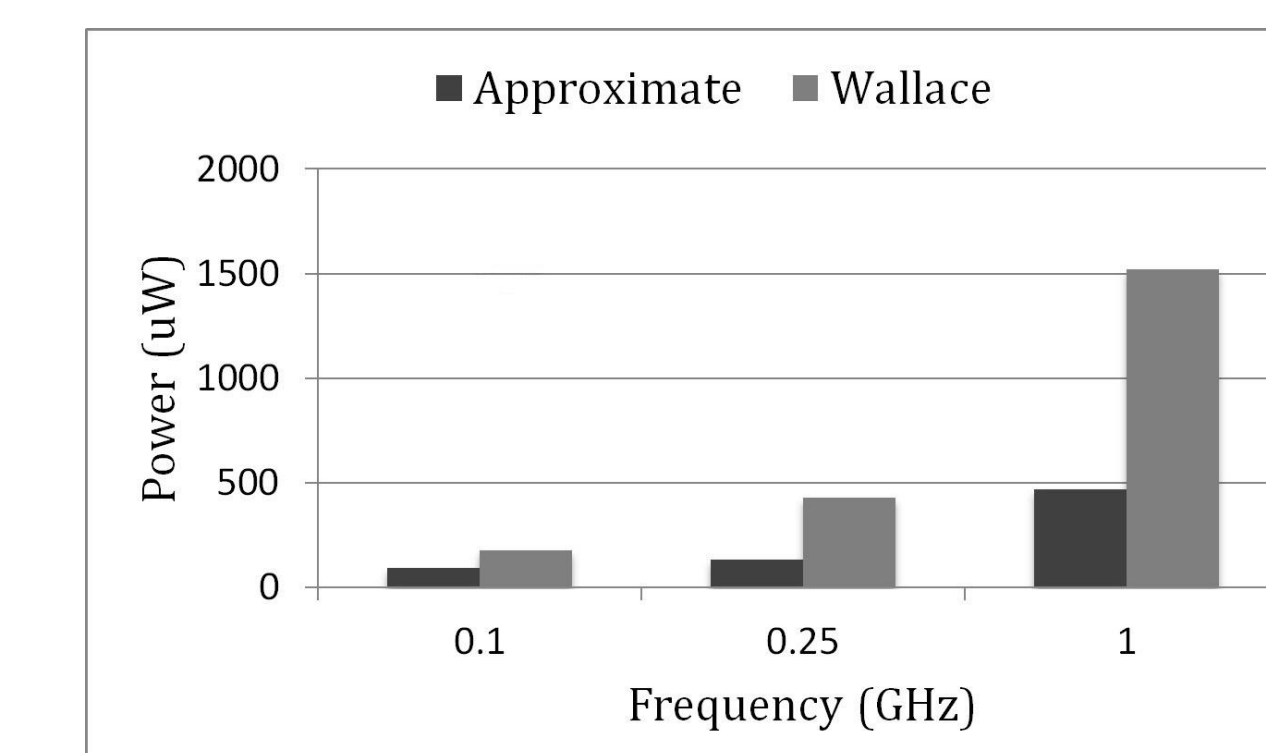
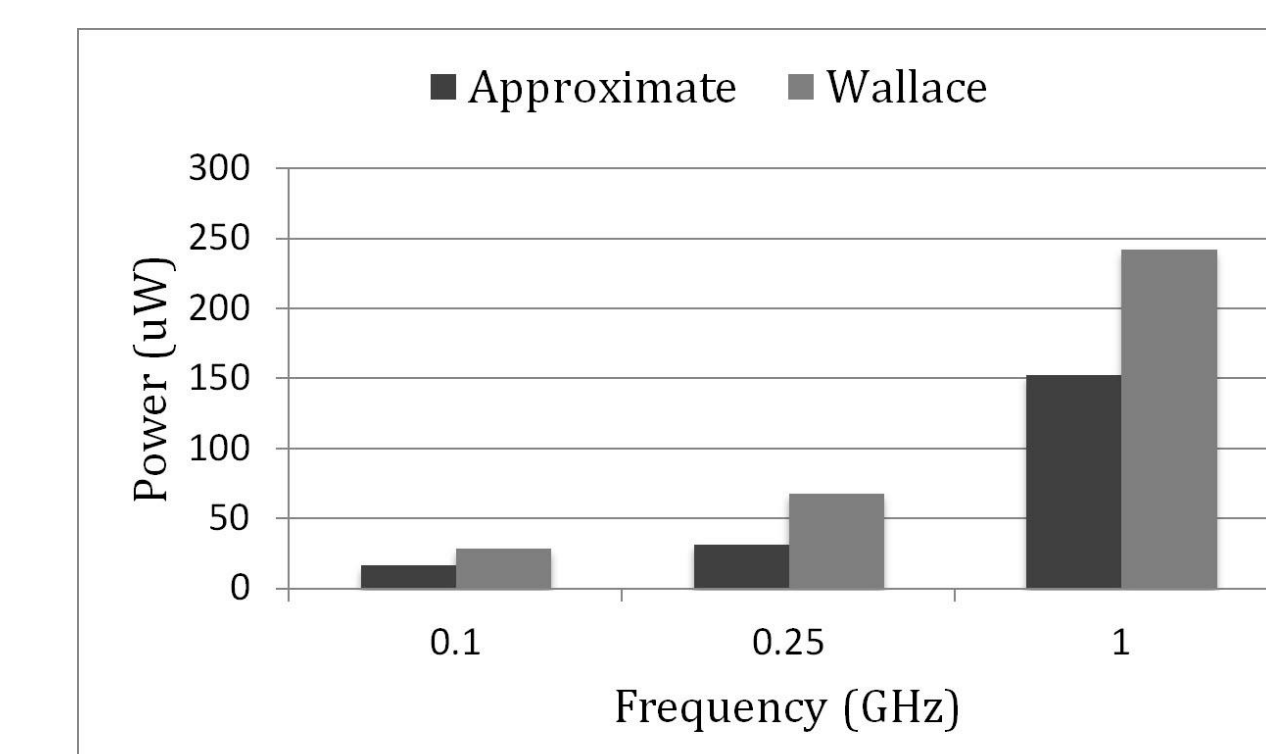


Fig. 5 Power consumption of  $8 \times 8$  (top) and  $16 \times 16$  (bottom) multipliers

## Accuracy Evaluation

- Metrics**
    - Error rate (ER)
    - Relative error distance (RED) and mean relative error distance (MRED)
- $$RED = \frac{|M' - M|}{M}$$
- $M$  - Accurate multiplication result  
 $M'$  - Approximate multiplication result
- Low ER and MRED are achieved if a proper number of bits for error reduction is used.

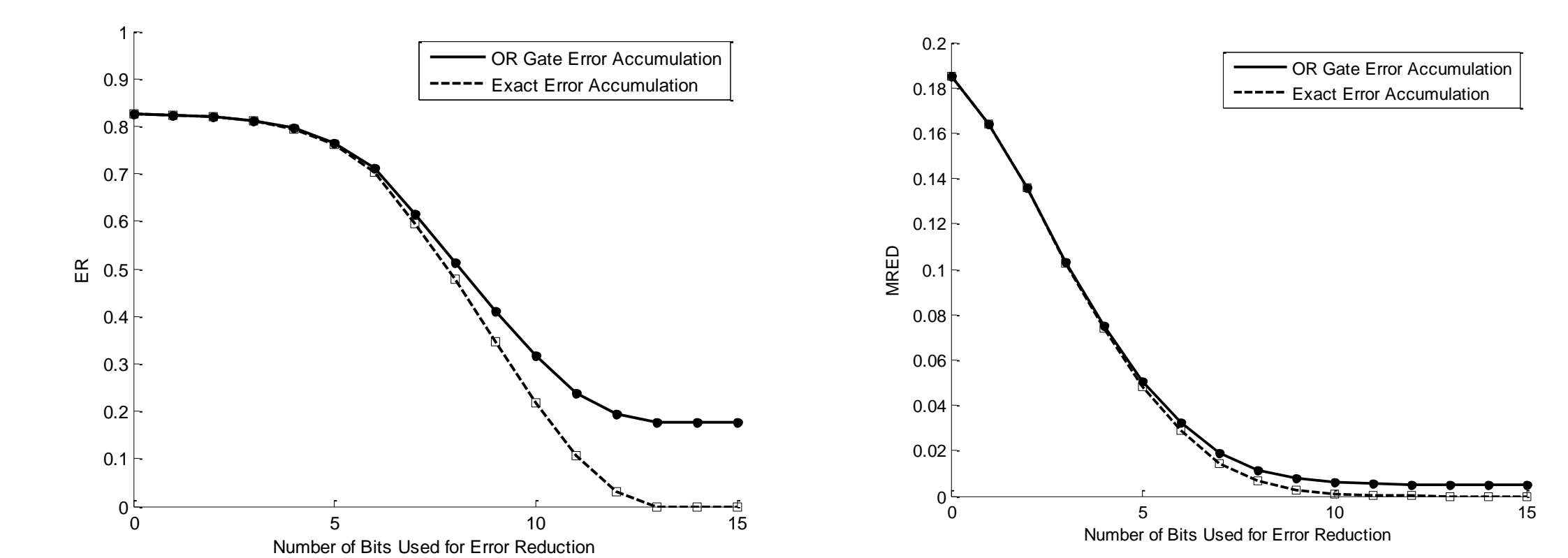


Fig. 6 ER and MRED of an  $8 \times 8$  approximate multiplier

- Image smoothing application**



Fig. 7 Image smoothing results using an approximate multiplier (left) and an accurate multiplier (right)

## Conclusion

- A novel approximate multiplier design is proposed using a newly designed approximate adder.
- On a statistical basis the proposed multiplier has a very small error distance and thus a high accuracy.
- The proposed design has a shorter critical path delay and a significantly lower power consumption compared to an exact Wallace multiplier.
- Current work investigates a new error accumulation scheme with higher accuracy and an accurate operation mode with full error recovery.

## References

- Han and Orshansky, 2013, IEEE ETS.
- Kulkarni *et al.*, 2011, IEEE Intl. Conf. on VLSI Design, pp. 346–351.
- Kyaw *et al.*, 2010, IEEE Intl. Conf. Electron Devices and Solid-State Circuits (EDSSC), pp. 1–4.
- Bickerstaff *et al.*, IEEE Symp. on Computer Arithmetic, pp. 33–39.

## Acknowledgements