DSDM2 Project review

Power analysis in approximate computing

Davide Galli

Approximate adders

The analyzed adders are:

- AMA *[1]*
- AXA *[2]*
- InXA *[3]*
- GeAr *[4]*

All of them have been compared with an exact RCA.

How it works: GeAr

GeAr breaks the full adder with some smaller sub-adders.

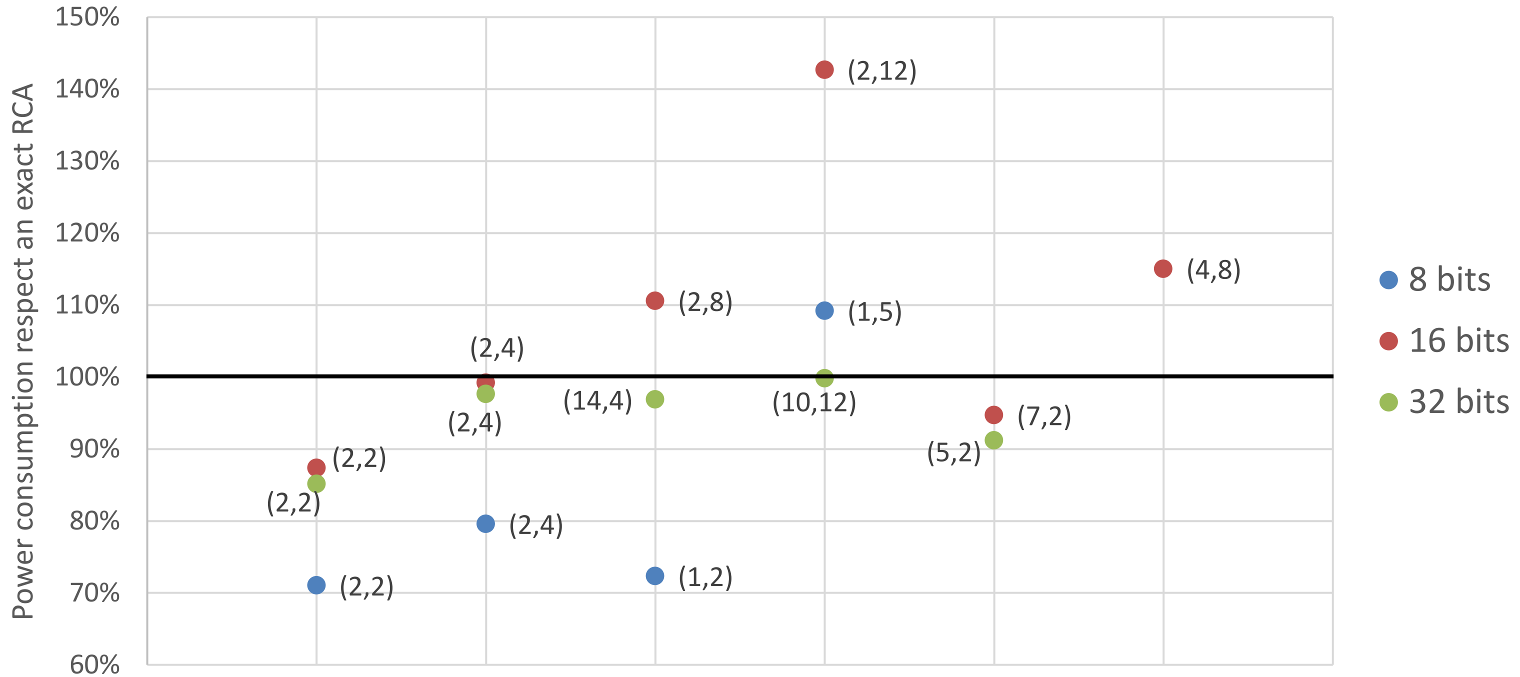
Sub-adders can be configured with two parameters:

- **R**: how many bits of the sub-adder will contribute to the final sum.
- **P**: how many *previous* bits are feed to the sub-adder to approximate the carry in.

The greater P, the more accurate the sum.

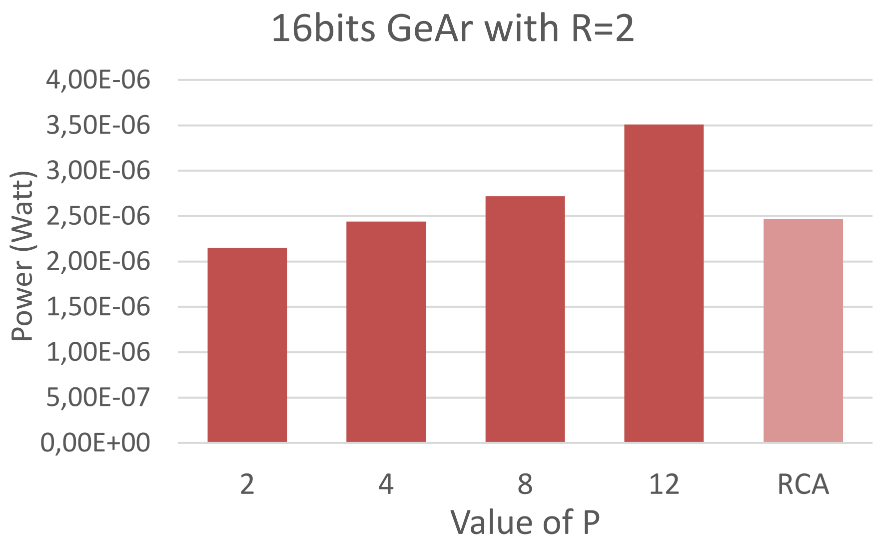
R+P is the bit width of each sub-adder.

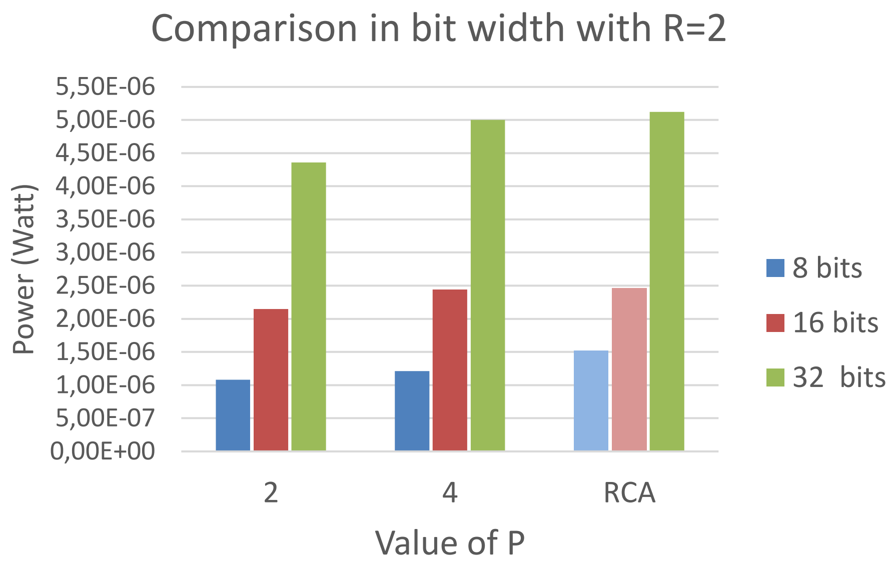
Results: GeAr



- ❖ The (2,2) configuration (R=2, P=2) is the best one for each input bit width.
  → It's better to have many small sub-adders.
- ❖ The smaller the input, the better the power consumption.

Results: GeAr





With constant R, as both P and input increase, power consumption increases.

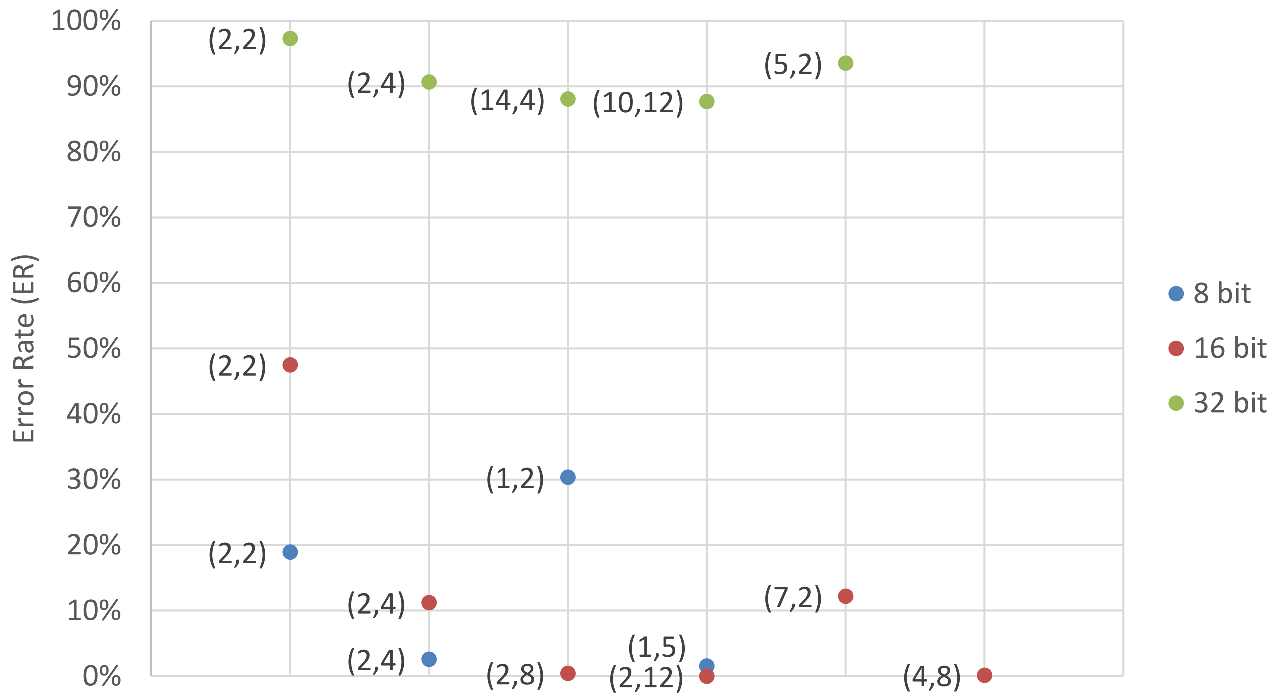
Results: GeAr - Extra

With the same number of sub-adders, the higher the value of P the higher the consumption.

Furthermore, the greater P the greater the size of the sub-adder itself.

BW	*R*	*P*	*# Sub-adders*	*Power Consumption (respect a RCA)*
8	2	2	3	71.05%
8	1	5	3	109.21%
16	2	12	2	142.68%
16	7	2	2	94.72%
16	4	8	2	115.04%

Results: GeAr - Extra



The error rate is shown for trading off between error and consumption.

The lower ER, the better.

How it works: AMA, AXA, InXA

These adders try to approximate each bit of the sum with various technics.

They can be configured based on how many LSBs to approximate and how many to leave exact.

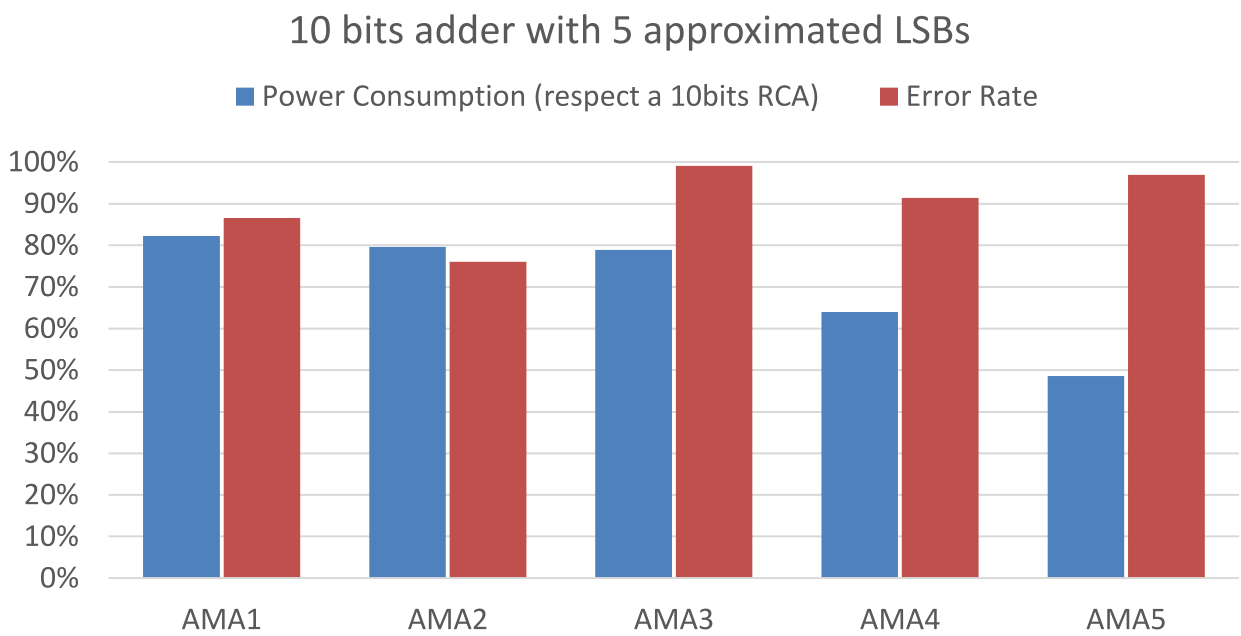
Results: AMA, AXA, InXA

- As approximate LSBs increase, consumption improves.
- When the approximate LSBs are fixed, if the bit width increases also the consumptions increase.
- Fixed the percentage of approximate bits (8/16, 5/10), the power reduction remains almost constant.

  → Unlike the RoBA multiplier

	BW	*Appr. bits*	*Power Reduction*
InXA1	10	5	23.68%
InXA2	10	5	8.55%
InXA3	10	5	23.68%
InXA1	16	8	25.61%
InXA2	16	8	7.32%
InXA3	16	8	22.36%

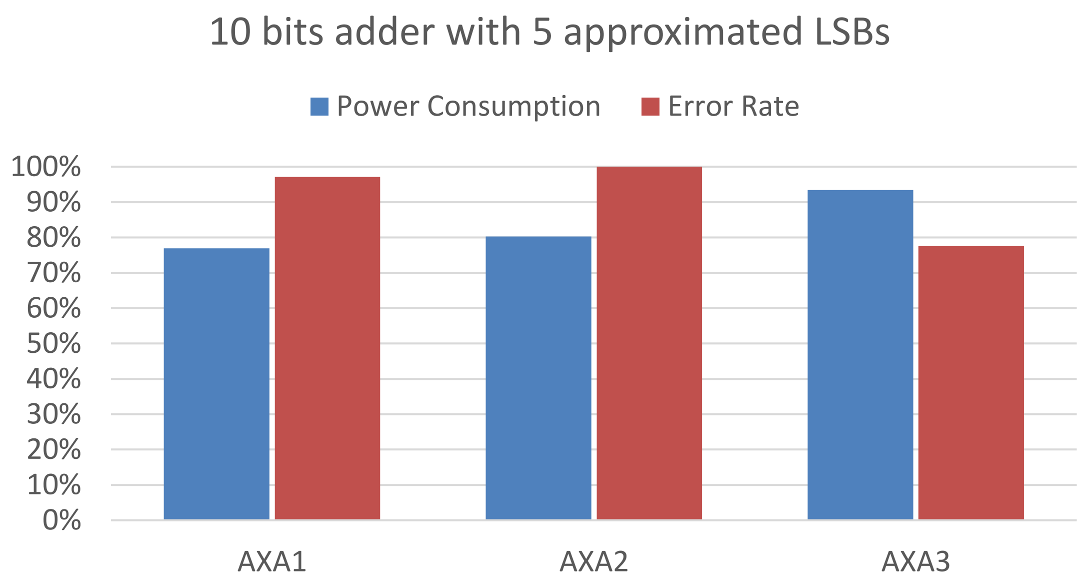
Results: AMA



The trend is mostly similar to the one in *figure 4* of [7], although the values are different ([6] consumes almost 20% less).

AMA5 has a completely different result: according to [7] it would have the worst consumption.

Results: AXA



The trend is mostly similar to the one in *figure 4* of [7], although the values are different ([7] consumes almost 10% less).

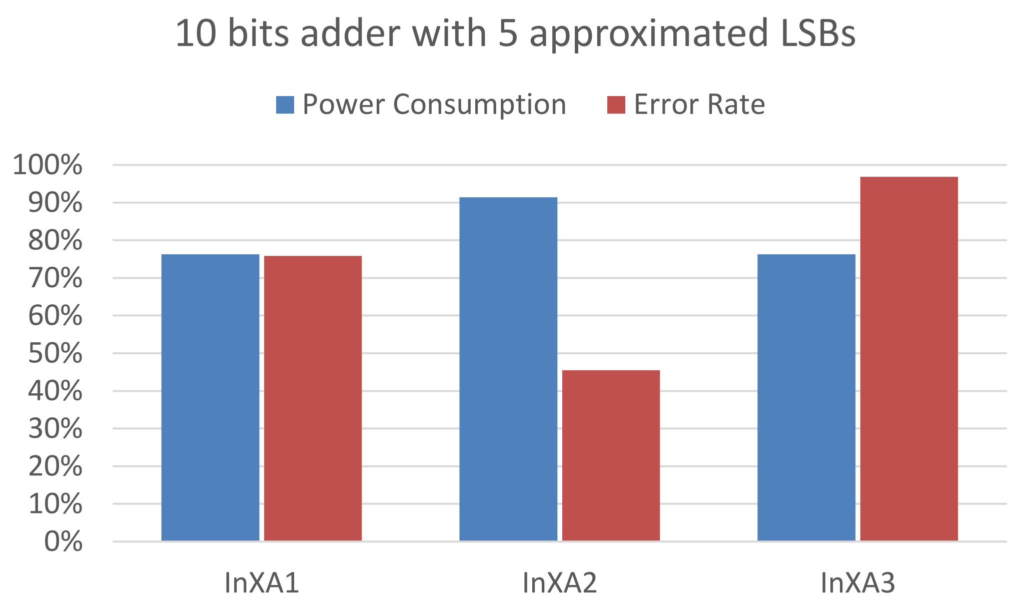
Results: AXA - Extra

AXA3 has an unexpected behavior:

Consumption on 16 bits with 5 approximate bits is better than those with 8 approximate bits in each area (internal, switching, leakage).

BW	*Appr. bits*	*Internal Power*	*Switching Power*	*Leakage Power*	*Total Power*
16	5	1.83E-14	1.15E-14	2.18E-06	2.18E-06
16	8	1.95E-14	1.41E-14	2.34E-06	2.34E-06

Results: InXA



The trend is mostly similar to the one in *figure 4* of [7], although the values are different ([7] consumes almost 20% less).

Approximate multipliers

The multipliers analyzed are:

- DRUM *[5]*
- RoBA *[6]*

Both have been compared with an exact multiplier.

(For these designs the ER is not shown because it is always 1)

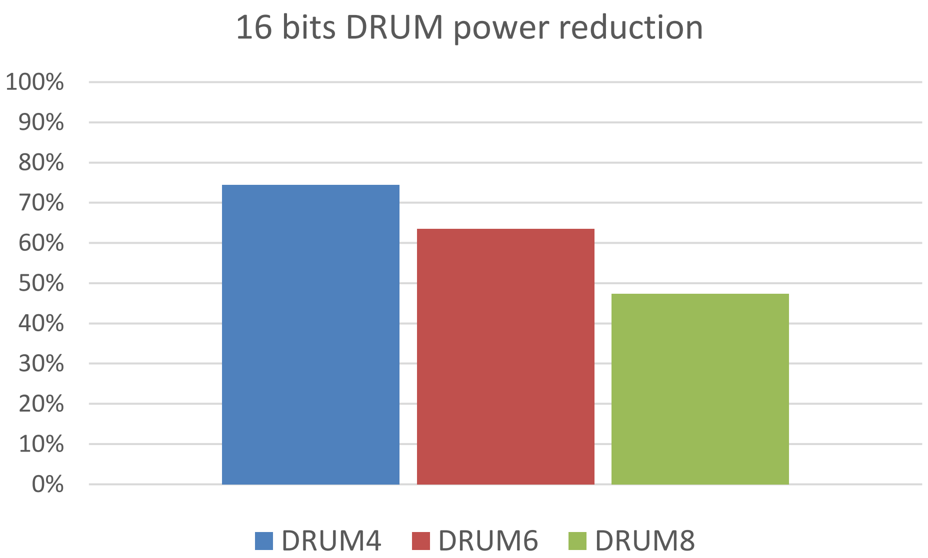
How it works: DRUM

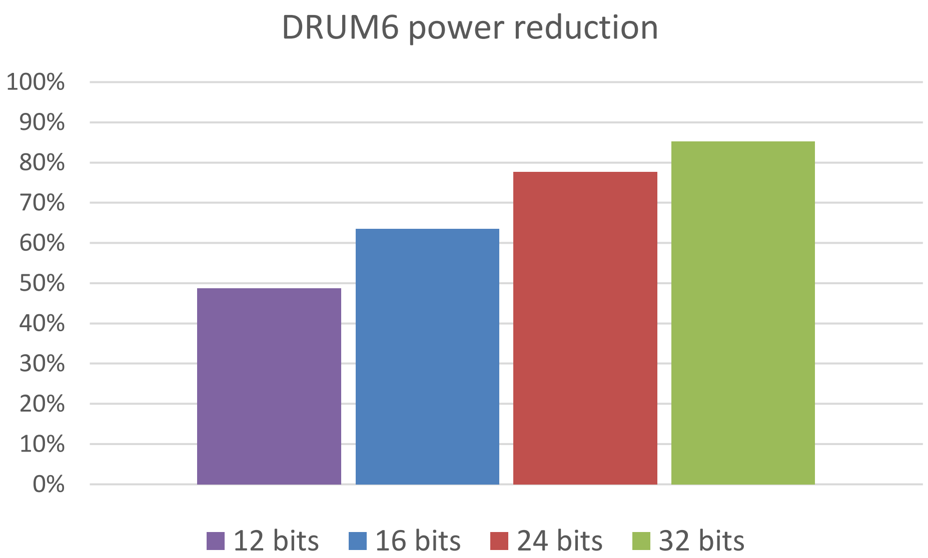
DRUM looks for the most significant one and cut the input by K bits from there down. The other LSBs are approximated.

Therefore, the real multiplication is K bits.

DRUM6 means K=6.

Results: DRUM





It has a <u>much higher </u>power reduction than all the other designs analyzed.

The trend is mostly similar to what is shown in the related paper [5] and in *figure 6* of [7], although the values are different.

Results: DRUM

The K parameter indicates by how many bits the actual multiplication will be.

Bit width	K	Power reduction
12	6	48.75%
16	8	47.37%
24	12	51.43%
32	16	56.17%

If K is half of the input width then the power consumption will be almost half as well (it is not extensively 50% because of the LOD, shifting and rounding components).

How it works: RoBA

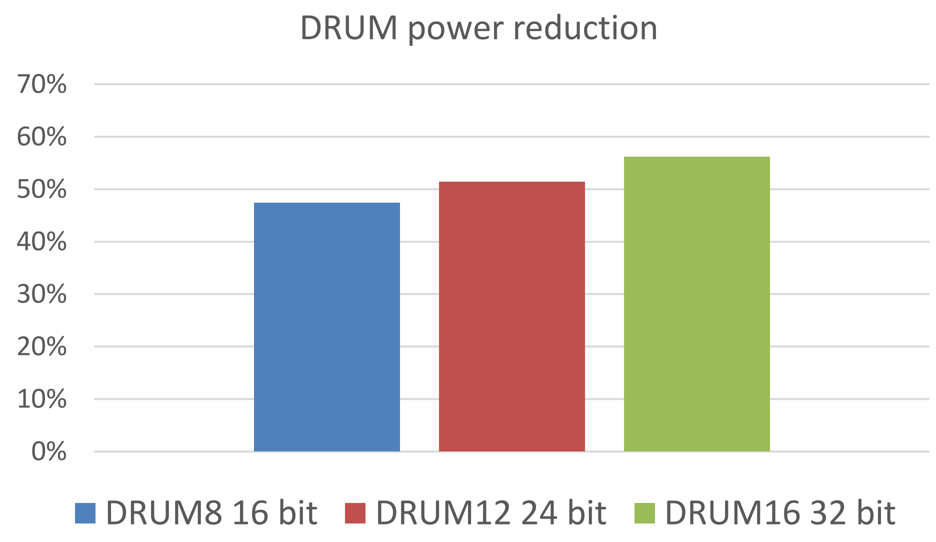
RoBA tries to approximate the inputs to the nearest exact power of 2, then the multiplication will be replaced by shifting and adding.

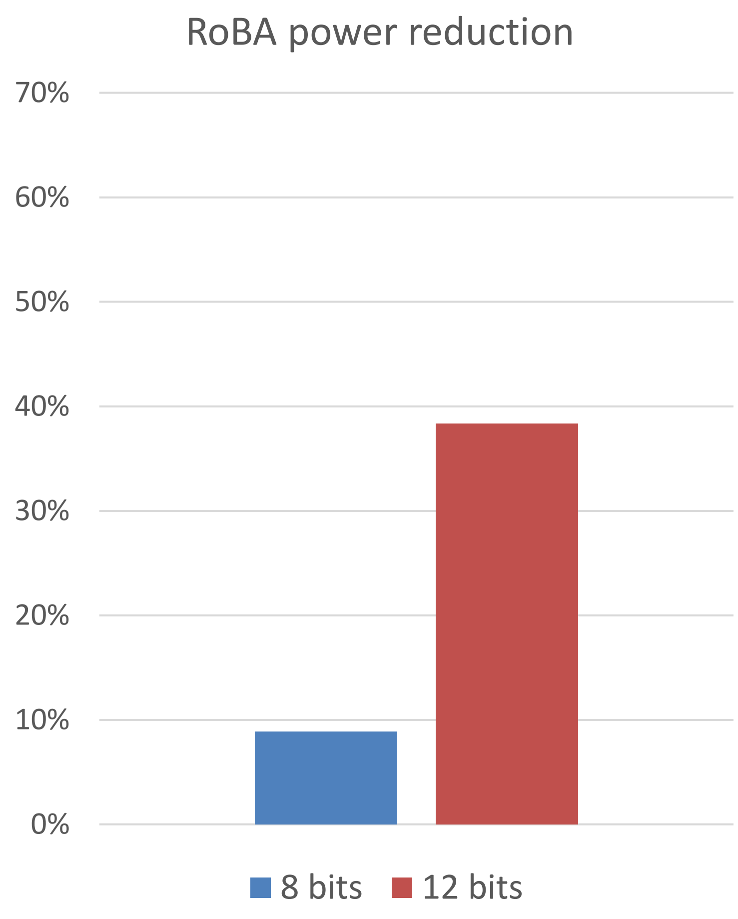
The only parameter that can be configured is therefore the input bit width.

Results: RoBA

Power reduction improves a lot as input size increases.

→ In the other designs it remained almost constant.





References

[1] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, “Low-Power Digital Signal Processing Using Approximate Adders,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 32, pp. 124–137, Jan 2013

[2] Z. Yang, A. Jain, J. Liang, J. Han and F. Lombardi, "Approximate XOR/XNOR-based adders for inexact computing," *2013 13th IEEE International Conference on Nanotechnology (IEEE-NANO 2013)*, Beijing, 2013, pp. 690-693.

[3] H. A. F. Almurib, T. N. Kumar and F. Lombardi, "Inexact designs for approximate low power addition by cell replacement," *2016 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Dresden, 2016, pp. 660-665.

[4] M. Shafique, W. Ahmad, R. Hafiz and J. Henkel, "A low latency generic accuracy configurable adder," *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, San Francisco, CA, 2015, pp. 1-6.

[5] S. Hashemi, R. I. Bahar and S. Reda, "DRUM: A Dynamic Range Unbiased Multiplier for Approximate Applications", *2015 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, Austin, TX, 2015, pp. 418-425.

[6] R. Zendegani, M. Kamal, M. Bahadori, A. Afzali-Kusha, and M. Pedram, “RoBa Multiplier: A Rounding-Based Approximate Multiplier for High-Speed yet Energy-Efficient Digital Signal Processing", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, pp. 393–401, Feb 2017.

[7] Deykel Hernández-Araya, Jorge Castro-Godínez, Muhammad Shafique and Jörg Henkel "AUGER: A Tool for Generating Approximate Arithmetic Circuits", *IEEE Latin American Symposium on Circuits and Systems* – LASCAS 2020, San José, Costa Rica, February 25 - 28, 2020.