INTEGRATION, the VLSI journal 55 (2016) 176–182

Contents lists available at ScienceDirect

INTEGRATION, the VLSI journal

journal homepage: www.elsevier.com/locate/vlsi

Energy efficient computing by multi-mode addition

Amir Albeck, Shmuel Wimer *

Bar-Ilan University, Engineering Faculty, Ramat-Gan, Israel

ARTICLE INFO

Article history:
Received 27 September 2015
Received in revised form
2 June 2016
Accepted 2 June 2016
Available online 10 June 2016

Keywords:
Adders
Low-power design
Multi-mode adders
Voltage scaling
Inaccurate computing

ABSTRACT

While adders are usually designed for the worst-case where their carry propagates through the entire bits, those cases rarely happen at real operation. This work takes advantage of the infrequent worst-case occurrences by designing adders for the average-case. Such design implies that computation errors may happen. Those are being corrected by implementing multi-mode addition with the aid of a dedicated control circuit. A power-delay-energy model is presented, enabling to find the optimal design point. We show that for cases where the system's critical paths are dictated by the adders, the system's operation voltage can be scaled, without harming the clock cycle and with very small performance degradation. For an adder per-se, potential energy savings of up to 50% is shown. The multi-mode adder has been integrated in a 32-bit pipelined MIPS processor, validating the correctness of such design approach.

1. Introduction

The design of adders assumes that addition must be accurately completed within a prescribed period, independently of the input operands. Consequently, the worst-case of carry propagation along the entire bits is targeted. When high performance is desired, prefix and carry look-ahead (CLA) adder architectures which consume large area and energy [1–3], are used. With the spread of mobile and green computing the focus of adder design is on low energy, which is what this paper is aiming at.

The most important factor in adder optimization is its *carry propagation probabilities*. It was used to estimate the adder's energy consumption in [4,5]. We present an architecture called *multi-mode*, where the adder is designed for the expected longest carry rather than the worst-case. The addition completes within a single clock cycle with minimum energy for most cases, while with a very small probability one or more cycles may be required.

Accuracy, performance and energy of adders were traded off by several works. Energy savings in adders can be achieved by sacrificing accuracy in applications that can tolerate it. Image processors for instance, can sometimes trade off the image quality for lower power. In [6] it was proposed to reduce the logic complexity of a full-adder at the transistor level and relax the numerical accuracy in a design of multi-bit adders. In addition to the inherent reduction in switched capacitance, the technique resulted in significant shortening of the critical paths, thus enabling voltage scaling. The approximate adder was used for video and image compression, achieving up to 69% power savings compared to accurate adders. Inaccurate computing has also been used for multiplication [7]. It achieved average power savings of 32–45% over corresponding accurate multiplier designs, with average errors of 1.4–3.3%.

Unlike [6] and [7] which pay in accuracy, a method called *Razor Design* [8] pays by small performance degradation. Voltage is scaled down so that most of the time the underlying logic safely completes its computation. The cases where the clock cycle was insufficient for a safe completion are detected during the execution. For those cases the computation is repeated by allocating more clock cycles. The seminal work in was followed by many razor-based works. A few to mention are [9] for 32×32 bit multi-precision Dynamic Voltage Scaling (DVS) multiplier, [10] for a loop-accelerator implementing an edge-detection algorithm, and [11] for programmable truncated Multiply and Accumulate (MAC) used for DSP applications.

Our method is somewhat reminiscent razor in the sense that it targets the average case of propagation delay rather than the worst-case. It is substantially different though. Firstly, razor indicates whether all the critical outputs of a logic stage (there may be many) have completed their proper evaluation within the given clock cycle, by augmenting the pipeline registers with supplementary latches and comparison logic. Our method is far more skim since it is tailored specifically for addition circuits. The detection takes place at the logic stage rather than at the sampling registers, as razor does. Secondly, razor does not modify the underlying logic circuits, since it is not aware of their particular structure. Our method in contrast, relies on the particular properties of carry propagation, leading to a change in the adder's architecture. Finally, razor gives up a little performance for power

* Corresponding author. Tel.: +972 3 5317208; fax: +972 3 7384051.
E-mail address: wimers@biu.ac.il (S. Wimer).

http://dx.doi.org/10.1016/j.vlsi.2016.06.002
0167-9260/

savings, but does not tolerate inaccurate computation. Our method in contrast allows inaccurate addition mode (with very low probability), for special purpose application that can tolerate such inaccuracies. This is elaborated in Section 6.

An adder operating in two modes was described in [12], taking advantage of the very small probability of long carries. It detected whether the operands comply with a "short" operation mode (most often), or a "long" operation mode is in order. The *short mode* enabled voltage scaling for power reduction. The *long mode* used two clock cycles to properly compute. Our adder proposal has higher power reduction potential for two reasons. Firstly, it uses three addition modes, and secondly, its design targets log n-bit carry propagation while [12] design targets $n/2$-bit carry propagation.

In an attempt to improve the overall throughput, a technique called *telescopic unit* [13] took advantage of data implications on the worst-case carry probability. The improvement is achieved by speeding up the clock signal, such that its cycle suffices for common input cases. Longer computations are split over several cycles. Being general and based on synthesis, the telescopic adder improves the throughput over a wide range of adder circuits. More improvements can however be achieved by taking advantage of the specific carry logic, as done in some of the above works.

D*ual-mode* adder probabilistic model was established in [14] and appropriate circuit architecture has been proposed to exploit the probabilistic nature of data. This paper extends the idea further into *multi-mode* by employing simple *carry-completion* control logic, presented in [2, Ch. 5.4]. We study thoroughly its energetic and performance advantages and show good matching between the theoretical energy-performance modeling and circuit simulations. It is shown how different processors architectures can leverage of multi-mode addition and how the probabilistic model matches with results obtained from real code running on a 32-bit MIPS processor.

The rest of the paper is organized as follows. Section 2 discusses the potential energy savings in multi-mode adders by voltage scaling. Section 3 presents carry probabilities implications on multi-mode addition. Section 4 models the energy consumption and energy-delay product at the transistor level. Section 5 compares the multi-mode adder with industrial design-ware adders. Section 6 describes multi-mode adder integration in a 32-bit MIPS pipelined processor and presents tests results. Usage in an industrial image processor is also discussed. We conclude in Section 7.

2. Energy saving in adders by voltage scaling

We consider a 64-bit carry propagate adder (CPA) designed in 65-nanometer TSMC process technology, operated in 1.3 V supply voltage. It is well-known that the expected longest carry propagation in n-bit addition is $O(\log_2 n)$ (see formal proof in [2, Ch. 5.3]). Rather than considering the carry propagation along 64 bits, we divide the adder into several k-bit groups. We found by SPICE simulations how far can the supply voltage be scaled, while a k-bit group still properly computes. The operands were set such that a carry-in pulse at the group's LSB will propagate through its entire bits.

Table 1 shows the lowest supply voltage yielding proper addition. The k-bit group comprises a pass-gate carry propagation chain combined with per-bit addition circuit [14]. The corresponding energies were obtained by integrating the current-voltage product over time. It is shown later how the multi-mode adder takes advantage of voltage scaling from 1.3 V to 0.95 V, yielding considerable energy reduction. Though implemented in 65 nm technology, a recently voltage scaling study in [15] shows a similar projection for 28 nm, where a good voltage scaling of 28-nanometer technology in the range of 1.0–0.7 V has been measured.

The separation of the bulk energy E in Table 1 into static and dynamic components matters. Those are later required to compute the energy consumed by an addition operation as a part of the entire computing system, where it is integrated in a pipelined processor. To this end a parameter $0<\lambda<1$ is introduced, counting the static energy λE and the dynamic energy $(1-\lambda)E$ portions of the bulk energy E.

3. Carry probabilities and multi-mode addition

Dual-mode adder architecture divides an n-bit CPA into $m=n/k$ independent groups of k-bit each [14], using a fast pass-gate carry-chain circuit architecture. As shown subsequently, the optimal values of k used in multi-mode adders does not exceed ten. For such size, fast circuit architecture, such as carry-skip or prefix-tree adders do not yield any speed advantage over pass-gate carry-chain.

It has been shown that the additions are properly computed with high probability within a single clock cycle. A dual-mode adder is working either in a single clock cycle, called *normal-mode*, or in m cycles, called *extended-mode*. A single clock cycle suffices when the longest carry does not exceed k bits. Otherwise, the adder will use more $m-1$ clock cycles to complete properly its computation. The probability p_{norm} that the longest carry does not exceed k bits at any of the m groups, a case where a single clock cycle suffices, is

$$p_{\text{norm}}(k,m)=\left(1-2^{-k}\right)^m=1-m2^{-k}+O\left(m^2 2^{-2k}\right)>1-m2^{-k}. \tag{1}$$

Unlike the dual-mode adder which spends m cycles for its extended mode, a multi-mode refines the extra cycles by using a simple carry-completion supplementary logic [2, Ch. 5.4]. Such logic validates when all the adder's bits have already been received their proper carry-in signal, thus ensuring proper addition result. We call *sub-normal* the mode where proper addition requires two clock cycles. The probability that a carry is either generated or killed within each of the $m/2$ $2k$-bit groups is

$$\left(1-2^{-2k}\right)^{\frac{m}{2}}=1-m2^{-(2k+1)}+O\left(2^{-4k}\right)>1-m2^{-(2k+1)} \tag{2}$$

The probability $p_{\text{sub\_norm}}$ of sub-normal mode to occur is therefore

$$\begin{aligned} p_{\text{sub\_norm}}(k,m)&=\left(1-2^{-2k}\right)^{\frac{m}{2}}-p_{\text{norm}}(k,m)\\ &=\left(1-2^{-2k}\right)^{\frac{m}{2}}-\left(1-2^{-k}\right)^m\cong m2^{-k}. \end{aligned} \tag{3}$$

Table 2 shows the probability $p_{\text{sub\_norm}}$ to take two clock cycles rather than one for proper addition, of an n-bit multi-mode adder comprising $m=n/k$ k-bit groups. It was obtained by simulations of random additions, yielding probabilities almost identical to (3).

It follows from (2) that the probability p_{ext} of the extended mode, where more than two cycles are required, is

$$p_{\text{ext}}(k,m)=1-\left(1-2^{-2k}\right)^{\frac{m}{2}}<m2^{-(2k+1)}. \tag{4}$$

Table 1
Group size versus supply voltage and energy.

Group size [bits]	8	16	32	64
Supply voltage [V]	0.553	0.655	0.841	1.3
Energy [10^{-12} J]	0.0103	0. 253	2.74	15.1