Trade-off between Energy and Quality of Service Through Dynamic Operand Truncation and Fusion

Wenchao Qian, Robert Karam and Swarup Bhunia
Case Western Reserve University, Cleveland, Ohio, USA
{wxq18,rak65,skb21}@case.edu

ABSTRACT

Energy efficiency has emerged as a major design concern for embedded and portable electronics. Conventional approaches typically impact performance and often require significant design-time modifications. In this paper, we propose a novel approach for improving energy efficiency through judicious fusion of operations. The proposed approach has two major distinctions: (1) the fusion is enabled by operand truncation, which allows representing multiple operations into a reasonably sized lookup table (LUT); and (2) it works for large varieties of functions. Most applications in the domain of digital signal processing (DSP) and graphics can tolerate some computation error without large degradation in output quality. Our approach improves energy efficiency with graceful degradation in quality. The proposed fusion approach can be applied to trade-off energy efficiency with quality at run time and requires virtually no circuit or architecture level modifications in a processor. Using our software tool for automatic fusion and truncation, the effectiveness of the approach is studied for four common applications. Simulation results show promising improvements (19-90%) in energy delay product with minimal impact on quality.

Categories and Subject Descriptors

B.8.1 [**Hardware**]: Performance and Reliability—*Reliability, Testing, and Fault-Tolerance*; C.5.4 [**Computer Systems Organization**]: Computer System Implementation—*VLSI Systems*; H.3.3 [**Information Systems**]: Information Search and Retrieval—*Information Search and Retrieval*

Keywords

Operation Fusion; Energy Efficiency; DSP; Quality of Service

1. INTRODUCTION

Energy efficiency has emerged as an important design constraint for embedded and portable electronics [1]. Design of energy-efficient systems can be accomplished by using design-time or run-time approaches. However, in general, they incur considerable performance overhead and often require circuit and architecture-level modifications. Truncation of input operand has been explored earlier in energy reduction [2]. However, they are applied at design time and are typically applicable to custom hardware. Memory based computing is another approach that provides an opportunity to reduce the computation energy through the use of LUTs [4].

In this paper, we propose a novel approach for improving energy efficiency in DSP and graphics applications through operation fusion and judicious truncation of operands. some applications are tolerant to output errors within an acceptable range based on the application requirements [2]. The proposed approach maps a complex fused operation into a lookup operation. Hence, it requires virtually zero design modifications and can be effective to reduce the number of operations. By ignoring a certain number of least significant bits, the LUT can be limited to an acceptable size, reducing the memory space and energy requirements while sacrificing the Quality of Service (QoS) or accuracy. We have presented the overall approach and proposed an efficient fusion process. We have developed a software tool to implement the fusion and truncation steps from the control and data flow graph, to create the LUT entries and to evaluate the trade-off between energy and accuracy. Using our tool, we have analyzed the effectiveness of the proposed approach for four representative applications. Our analysis shows 19 - 90% improvement in energy efficiency with only a modest loss of quality.

2. OVERALL APPROACH

The overall approach is described as follows. At design-time, input applications are first examined for opportunities for fusion. Once suitable operations are fused, the corresponding LUT content are generated. During run time, on-demand truncation schemes are applied to the complex fused functions to improve the energy efficiency and memory space. After a specific scheme is applied, the required LUTs are loaded into memory, and LUT operations is executed accessing memory to the correct address to retrieve the result. When finished, the QoS requirements are re-evaluated, and the truncation scheme is adjusted as needed.

Fusion incorporates three routines: (i) fusion of random LUT-based operations, (ii) fusion of bit-sliceable operations, and (iii) fusion of custom datapath operations. We have developed a heuristic for partitioning a target application

GLSVLSI'14, May 21–23, 2014, Houston, Texas, USA.
ACM 978-1-4503-2816-6/14/05.
http://http://dx.doi.org/10.1145/2591513.2591561.

into multi-input multi-output operations. The vertices inside each partition are fused to form a single vertex to be mapped as a LUT operation [3]. LUT implementation is enabled by truncating some insignificant bits from input operands, if they have large bitwidth (than a threshold), otherwise the LUT size will grow exponentially.

The memory-based computing paradigm calls for storing not only the data itself, but also the LUT content, in the same memory space. By involving the OS in the LUT storage process, the system is run-time configurable using the address containing the LUT content. In address generation, the base address represents the operation type, while the operands are combined in a shuffled pattern to form the offset address. This allows a lookup of a specific function output given by the two input operands. The operands form two different offset address, but use the same base address, as they are both inputs to the same function type. The LUT content needs to be computed and loaded only one time, further improving the energy required for repeated execution of the same kernel.

Loading the correct values, however, is not trivial, since any change into the number of truncated bits results in different offset addresses for given inputs. We store values in the reverse order, with the truncations being done from the MSBs. Hence, truncating each bit will disable part of the LUT content. In other words, useful entries will be stored in consecutive blocks, making the memory management trivial.

For applications that can tolerate a small amount of output error, input and/or output truncation becomes an attractive option for reducing the lookup table size and access energy, since the LUT size varies exponentially with the input bitwidth. Truncating few LSB bits often allow the output value to remain more or less the same, thus inducing graceful degradation in QoS.

The proposed approach is amenable for automation, and we have developed a software tool that can automatically identify the operations to fuse, perform appropriate operand truncation, create LUT content, and replace a cluster of fused operations into one lookup operation.

3. SIMULATION RESULTS

We have implemented and simulated several representative applications from the DSP and graphics domains, including a two-dimensional discrete cosine transform, color interpolation, finite impulse response filtering, and the discrete wavelet transform. These applications are compute-intensive, and may potentially benefit from LUT implementations; but more importantly, they can tolerate small amount of output error. The baseline implementation used a 16-bit adder for addition, a 16-bit lookup table for multiplication, and an operating frequency of 500MHz. Energy values were obtained using Synopsys Design Compiler for 32nm process models, while the memory access parameters were generated with CACTI. Finally, we assumed that all lookup table and datapath operations are completed in a single cycle. For a certain output bit width, LUT energy values are the same for different input sizes. This is because we assume that the total cache size does not change and run-time reconfiguration is being used to write the proper size of LUT content into cache according to input and output bit width. We evaluate the output quality using Peak Signal-to-Noise Ratio (PSNR) for color interpolation and DCT, and Mean Squared Error (MSE) for DWT and FIR (uniform and preferential truncations). Simulation results in Fig. 1 show promising improvements (19-90%) in energy delay product with a minimal impact on quality.

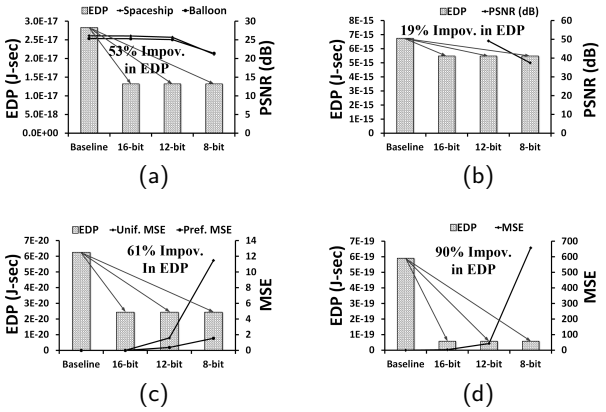

Figure 1: Comparison of EDP-quality for: (a) 2D-DCT, (b) color interpolation, (c) FIR, (d) DWT.

4. CONCLUSION

We have presented fusion approach enabled by judicious operand truncation that allows trade-off between energy efficiency and quality for signal processing and graphics applications. It utilizes the embedded memory of a processor to realize lookup-based evaluation of a fused operation. Thus, the proposed approach requires virtually no design modifications and can be used to trade-off energy versus accuracy at run-time. To minimize the quality impact associated with truncation, we have presented a preferential truncation strategy that optimally chooses the bits to truncate for an application. The approach is amenable to automation, and we have developed a software tool to evaluate it for four common applications. The proposed approach can be easily combined with existing run-time energy management schemes such as voltage scaling or adaptive body biasing.

5. ACKNOWLEDGEMENTS

The research has been funded in part by National Science Foundation (NSF) Grants #1002090 and #0964514.

6. REFERENCES

[1] N. Banerjee, G. Karakonstantis and K. Roy, "Process Variation Tolerant Low Power DCT Architecture," *DATE*, 2007.

[2] K. Kunaparaju, S. Narasimhan and S. Bhunia, "VaROT: Methodology for Variation-Tolerant DSP Hardware Design Using Post-Silicon Truncation of Operand Width," *VLSI Design*, 2011.

[3] J. Cong and S. Xu, "Technology mapping for fpgas with embedded memory blocks," *FPGA*, 1998.

[4] S. Paul and S. Bhunia, "Dynamic Transfer of Computation to Processor Cache for Yield and Reliability Improvement," *IEEE Trans. on VLSI systems*, 2011.