Simplified Floating-Point Units for High Dynamic Range Image and Video Systems

Javier Hormigo and Julio Villalba
Dept. Computer Architecture
Universidad de Malaga
Malaga, Spain
fjhormigo@uma.es

***Abstract*—The upcoming arrival of high dynamic range image and video applications to consumer electronics will force the utilization of floating-point numbers on them. This paper shows that introducing a slight modification on classical floating-point number systems, the implementation of those circuits can be highly improved. For a 16-bit numbers, by using the proposed format, the area and power consumption of a floating-point adder is reduced up to 70% whereas those parameters are maintained for the case of a multiplier.**

Keywords—Floating-point units; High Dynamic Range imaging; video; consumer electronics; optimization.

I. INTRODUCTION

The field of High Dynamic Range (HDR) imaging and video is an emergent trend in consumer electronics. Although human eyes can cover a luminance range of more than 10 orders of magnitude (5 orders simultaneously), conventional videos and images can only cover a range of about 2 [1]. Thus, most of the HDR coding standards use floating-point numbers for pixel representation. For example, OpenEXR typically utilizes 16-bit floating-point numbers, i.e., a Half-Precision IEEE754-like format, to represent the color components [2].

In the past several years, many HDR applications have been developed, ranging from generation [3], tone mapping, enhancement, etc. Along with the codification of HDR pixels, most of those HDR applications require the use of floating-point number computation. Thus, the new devices created to support HDR systems have to deal with floating-point arithmetic. Any improvement on implementing floating-point computing will directly benefit those systems for HDR images and videos. That is especially important for video applications since high computation capabilities are required for real time.

Recently, a new fixed-point representation system has been utilized in [5] to optimize the FPGA implementation of FIR filters. In [6], this representation system, which is called Half-Unit Biased (HUB) format, is formalized and extended to floating-point numbers. In this paper, we show that the use of HUB formats allows simplifying arithmetic units dealing with real numbers, which may facilitate the hardware implementation of HDR video systems. The new format reduces area and power consumption maintaining the accuracy.

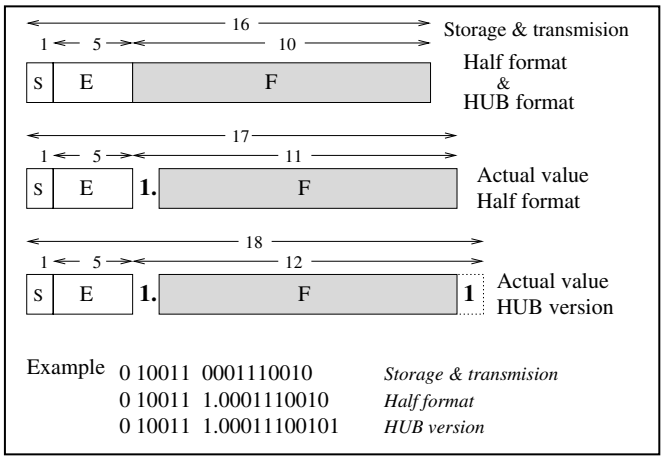

Fig. 1. Comparison between Half (OpenEXR) and HUB formats.

II. HUB FLOATING-POINT FORMAT

The HUB formats are based on a simple modification of conventional ones. This modification allows the rounding of the results of arithmetic operations simply by truncation, while it maintains the same precision. Taking into account that the rounding circuit represents an important portion of floating-point units, its elimination means a significant simplification of those units.

For floating-point formats, the HUB representation system is based on appending an implicit Least Significant Bit (LSB), which always equals one, to the mantissa of the floating point number. In this way, the values exactly represented under that format are shifted, such as the elimination of the LSBs of a number produces a round-to-nearest. This appended LSB does not have to be stored or transmitted. It only has to be taken into account when an operation is performed. Thus, the new format has the same numbers of explicit bits and precision as the original one. Fig. 1 shows an example of this transformation for the half format used on OpenEXR.

III. SIMPLIFYING ARITHMETIC UNITS

The finalization of any floating-point operation generally requires rounding the mantissa of the result to fit into the format size. The round-to-nearest ties-to-even is the preferred

This work was supported in part by the Ministry of Education and Science of Spain and Junta of Andalucía under contracts TIN2013-42253-P and TIC-1692, respectively, and Universidad de Málaga, Campus de Excelencia Internacional Andalucía Tech .

rounding mode for floating-point arithmetic and the only one implemented on OpenEXR [2]. The implementation of this rounding generally requires the computation of the sticky bit, some logic to decide the direction of the rounding, and a final addition. Thanks to the use of HUB formats all this hardware is prevented, since the rounding is performed simply by discarding the LSBs.

In contrast, the hidden LSB has to be explicitly used to perform the actual arithmetic operation. Although this LSB is a constant value, it may produce an area increase of the circuit which performs the mantissa fixed-point arithmetic operation. However, this area increase is normally compensated with the saves due to the elimination of the rounding circuit.

The floating-point addition requires several modules: the mantissa alignment, the mantissa addition/subtraction, the normalization and the rounding. Using HUB formats, the rounding is eliminated which saves area and delay. Furthermore, despite the included LSB, the size of the fixed-point adder is also reduced, since no guard bits are required for rounding. Thus, the improvement is very significant.

For classic floating-point multiplier, the alignment is not required and normalization is much simpler. Again, if the HUB format is used, the multiplier does not require the rounding module including the sticky bit computation. However, the fixed-point multiplier is augmented due to the extended LSB. Thus, depending on the specific parameters the overall area may be reduced or increased.

IV. Implementation Results

To measure the benefits of using the HUB format, an adder and multiplier have been described in VHDL and implemented targeting a 65-nm CMOS standard cell library technology. Both formats, half-precision format (STD) and the corresponding HUB format, have been implemented targeting several clock frequencies, from 50 to 950 MHz. Fig. 2 shows the area required for the adder and multiplier. It is clearly seen that the proposed adder requires significantly less area than the standard one, especially when the clock frequency goes up. Specifically, the use of the proposed format produces an area reduction which ranges between 32% and 71% (more than three times less). However the area of the multipliers is very similar for both formats.

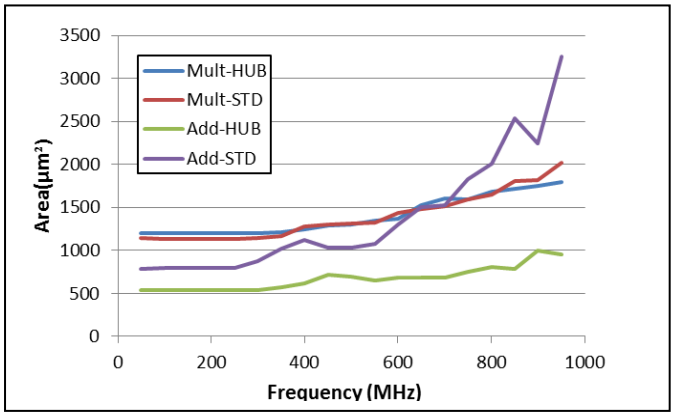

Fig. 2. Area requeriments of different floating-point units.

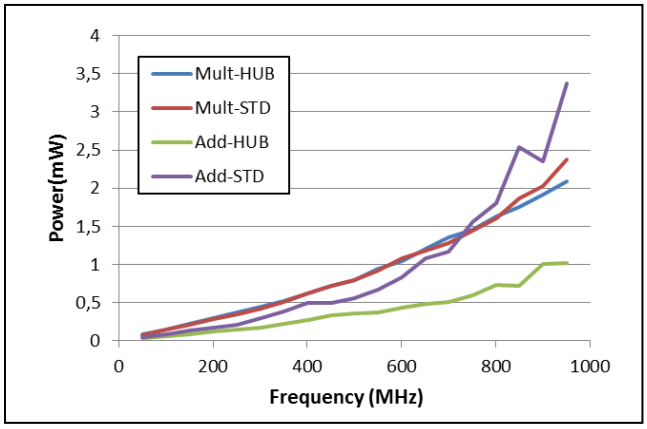

Fig. 3. Power consumption of different floating-point units.

Similarly, the power consumption of these circuits is shown in Fig. 3. As expected, the power consumption is also dramatically reduced for the adder circuit, while remaining very similar for the multiplier circuit. The consumption improvement ranges from 30% to 72% for the adder but only from -6% to 11% for the multiplier.

Taking into account that a typical application has at least the same amount of additions as multiplications and HUB circuits have the same accuracy as standard one, we can conclude that the use of the HUB format on these applications can significantly reduce the area and power consumption.

V. Conclusions

In this paper we propose the use of floating-point HUB formats for HDR image and video applications. Using these formats, the same accuracy is kept, but rounding circuits are practically eliminated. Consequently, arithmetic operations may be dramatically simplified. As an example, we have shown that for 16-bit floating-point numbers, conventional adders demand more than three times as much area and power consumption as HUB adders, when high speed is required. In contrast, the improvement for multipliers is much more moderate for high frequencies and even the opposite for low frequencies. However, significant savings are expected for typical applications. A few patent applications have been filed for different issues regarding to the circuits to operate under the new format.

References

[1] M. Ali, T. Ai, A. Gill, J. Emilio, K. Ovtcharov, and S. Mann, “Comparametric HDR (High Dynamic Range) imaging for digital eye glass, wearable cameras, and sousveillance,” in Technology and Society (ISTAS), 2013 IEEE International Symposium on, pp. 107–114, June 2013.

[2] Florian Kainz, Rod Bogart, and Piotr Stanczyk,”Technical Introduction to OpenEXR”, http://www.openexr.com, 2013.

[3] Young-Su Moon; Jonghun Lee; Yong-Min Tai; Junguk Cho; Do-Hyung Kim; Shi-Hwa Lee, "A ghost-free pseudo-multiframe HDR," Consumer Electronics (ICCE), 2014 IEEE International Conference on., pp. 260–261, Jan. 2014.

[4] J. Hormigo and J. Villalba, “Optimizing DSP circuits by a new family of arithmetic operators,” in Signals, Systems and Computers, 2014 Asilomar Conference on, pp. 871–875, Nov 2014.

[5] J. Hormigo and J. Villalba, “New Formats for Computing with Real-Numbers under Round-to-Nearest”, Computers, IEEE Transactions on, “unpublished”.