

III. PERFORMANCE EVALUATION

The proposed DF hardware architecture is verified with Verilog RTL simulations and the results are conformed to JM15.1 reference software. The Verilog code is then synthesized to Xilinx XC2V8000 Virtex II device. The power consumptions for signal, logic and clock categories of our DF hardware implementations on FPGA are then estimated by using Xilinx ISE 8.2i XPower tool. We take our power estimation with Parlak's work [5] for fair comparisons and both of them are under the same experimental conditions (Clock rate at 50 MHz and both use block selectRAMs for internal memory). Table I lists the power comparisons of our design with [5], which has two different hardware architectures. It clearly shows that our proposed DF consumes less power than [5] in all categories. Also, in Parlak's architecture, 5248/5376 processing cycles are required to filter an MB. In our scheme, only 100-196 cycles/MB are required. The possible reason is that [5] adopted 8-bits data bus to access each pixel while our design utilized 32-bits wide data bus to access 4 pixels each time.

TABLE I
POWER ESTIMATION COMPARISONS WITH [5]

Category	[5]DBF_4x4 HW	[5]DBF_16x16 HW	Proposed	Reduction
Clock	56.37 mW	50.36 mW	46.63 mW	3.73 mW
Logic	145.65 mW	52.47 mW	21.10 mW	31.37 mW
Signal	83.56 mW	79.39 mW	63.80 mW	15.59 mW
Total	285.58 mW	182.22 mW	131.53 mW	50.69 mW

DF uses BS to determine the appropriate strength of the filter applied on the edge. The percentage for BS=0 in P frames from our simulation results occupies the most portions of the distributions. When BS=0, filtering processes can be skipped and saved. The proposed HESPA takes this advantage of this fact and benefit from its property. Table II lists three categories of power estimation and compares the power reduction between HESPA being on and HESPA being off. Our proposed HESPA can save up to one-third (34%) power consumption in Logic and Signal and thus can speed up DF processing significantly.

Our DF hardware performance is also evaluated by comparing with various state-of-the-art designs. As shown in Table III, our design requires less Transposition Buffers and fewer gate counts than [3] which has the similar design approach with ours. Although our design requires more processing cycles than [4], we can save more gate counts and Transposition Buffer usages comparing with [4]. That is because Tobajas et al. [4] proposes a double filter with two identical filtering units opposing to our 1-D filtering strategy. Moreover, our proposed HESPA can achieve only 100 cycles per MB in the best case, which even outperforms the 2-D architecture in [4] (110 cycles). Our design consumes 19.8 K gates at a clock frequency of 200 MHz under SMIC 0.18μm standard cell library. The hardware cost of our proposed scheme is very competitive compared with other state-of-the-art literatures using 1-D filtering architecture.

TABLE II
POWER COMPARISONS BETWEEN HESPA ON/OFF

Sequence	100 frames of Forman QCIF		
Category	HESPA OFF	HESPA ON	Reduction
Clock	46.63 mW	46.63 mW	0
Logic	21.10 mW	13.90 mW	34.12%
Signal	63.80 mW	42.04 mW	34.11%
Whole FPGA	624.36 mW	579.08 mW	7.25%

TABLE III
COMPARISONS WITH STATE-OF-THE-ART LITERATURES

Ref	Filter Pipeline	1-D / 2-D	Processing Order	RAM type	RAM size	Transpose Buffer(4x4)	Tech	Gate Count	Processing cycles /MB
[2]	non-pipeline	1-D	sequential basic	two-port	2x80x32	4	0.25 um	20.66k	614
[3]	5-stage pipeline	1-D	hybrid	single-port	2x96x32 / 2Nx32	7	0.18 um	21.492k	204
[4]	5-stage pipeline	2-D	hybrid	single-port	64x32	8	0.18 um	12.6k(*)	110
[5]	non-pipeline	1-D	hybrid	two-port	1792x8	none	FPGA	5.3k(*)	5248 / 5376
[6]	non-pipeline	1-D	hybrid	two-port	16x32	2	0.25 um	13.41k	300
proposed	4-stage pipeline	1-D	hybrid	two-port	384x32	5	0.18 um	19.8 k / 9.9k(*)	100 ~196

NOTE: (*) GATE COUNT EXCLUDING ITS TRANSPOSITION BUFFER.

IV. CONCLUSIONS

This work proposes a low power DF architecture in H.264/AVC. The proposed HESPA offers an intelligent edge skip aware mechanism in filtering horizontal edges. The contribution of this work is to propose an efficient DF hardware architecture that can adaptively reduce power consumption up to 34% for the P frame of QCIF video sequence and saves the filtering orders up to 100 clock cycles per MB. The gate count is only 19.8K with 0.18μm cell library. The proposed edge filtering order, together with the proposed four-stage pipeline and memory update scheme, help to reduce required memory bandwidth and raise our throughput to real-time filtering for 1080HD video at 30 fps with a 70MHz clock frequency.

REFERENCES

- [1] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz, "Adaptive deblocking filter," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 13, no. 7, pp. 614-619, July 2003.
- [2] Y. W. Huang, T. W. Chen, B. Y. Hsieh, T. C. Wang, T. H. Chang, and L. G. Chen, "Architecture design for deblocking filter in H.264/JVT/AVC," in *Proc. IEEE Int. Conf. Multimedia Expo.*, July 2003, vol. 1, pp. 693-696.
- [3] K. Xu and C. S. Choy, "A Five-Stage Pipeline, 204 Cycles/MB, Single-Port SRAM Based Deblocking Filter for H.264/AVC," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 18, no. 3, pp. 363-374, Mar. 2008.
- [4] F. Tobajas, G. M. Callico, P. A. Perez, V. de Armas, and R. Sarmiento, "An Efficient Double-Filter Hardware Architecture for H.264/AVC Deblocking Filtering," *IEEE Trans. Consum. Electron.*, vol. 54, no. 1, pp. 131-139, Feb. 2008.
- [5] M. Parlak and I. Hamzaoglu, "Low power H.264 deblocking filter hardware implementations," *IEEE Trans. Consum. Electron.*, vol. 54, no. 2, pp. 808-816, May 2008.
- [6] C. C. Cheng, T. S. Chang, and K. B. Lee, "An in-place architecture for the deblocking filter in H.264/AVC," *IEEE Trans. Circuits Syst. II*, vol. 53, no. 7, pp. 530-534, Jul. 2006.