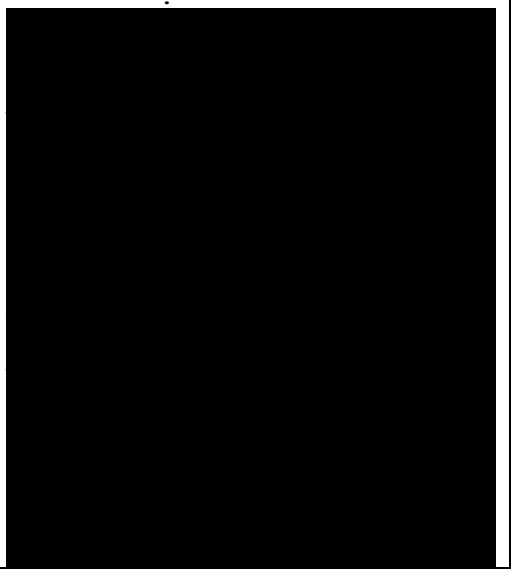
High Throughput VLSI Architecture for HEVC SAO Encoding for Ultra HDTV 6 h 14 Abstract—This paper presents a high performance, silicon area efficient, and software configurable hardware architecture for sample adaptive offset (SAO) encoding. The paper proposes a novel	Kiến trúc VLSI lưu lượng cao để giải mã HEVC SAO cho HDTV siêu cao Tóm tắt-Bài báo này trình bày kiến trúc phần cứng hiệu suất cao, hiệu quả về mặt diện tích đế bán dẫn và có thể cấu hình được bằng phần mềm để giải mã độ lệch tương thích mẫu (SAO). Bài báo đề xuất một kiến trúc

architecture consisting of single largest coding unit (LCU) stage SAO operation, unified data path for luma and chroma channels, add-on external interfaces on frame level statistics collection units to allow fine control over the parameter estimation process, flexible rate control and artifact avoidance algorithms. The unified data path consists of 2D-block based processing with 3 pipeline stages for statistics generation and multiple offset rate-distortion cost estimation blocks for high performance. The proposed design after placement and routing is expected to take-up approximately 0.15 mm2 of silicon area in 28nm CMOS process. The proposed design at 200 MHz supports 4K Ultra HD video encoding at 60fps. Simulation experiments have shown average bit-rate saving of up to 4.3% with in-loop SAO filtering and various encoder configurations.	mới bao gồm phép toán SAO tầng đơn vị mã hóa lớn nhất (LCU), đường dữ liệu hợp nhất cho các kênh luma và chroma, các giao diện bên ngoài add-on ở bộ thu thập thống kê mức khung để điều khiển tinh quy trình ước lượng tham số, điều khiển tốc độ linh hoạt và các thuật toán tránh nhiễu. Đường dữ liệu hợp nhất bao gồm quá trình xử lý dựa vào khối hai chiều với ba tầng pipeline để tạo thống kê và nhiều khối ước lượng chi phí tỷ lệ offset-méo. Pipeline: cấu trúc đường ống Thiết kế đề xuất sau khi đặt và định tuyến dự tính sẽ chiếm khoảng 0.15 mm2 diện tích đế bán dẫn trong quá trình CMOS 28nm. Thiết kế đề xuất ở tần số 200 MHz hỗ trợ cho quá trình giải mã video HD siêu cao 4K ở 60fps. Các thí nghiệm mô phỏng cho thấy đã tiết kiệm được tốc độ bit trung bình lên đến 4.3% với các cấu hình lọc SAO trong vòng lặp và các cấu hình giải mã khác nhau.
I. INTRODUCTION High Efficiency Video Coding (HEVC) is the next generation video compression standard ratified in January 2013, which was jointly developed by ISO/IEC and ITU-T [1]. The new standard promises nearly 50% bit-rate saving as compared to existing ISO/IEC and ITU-T standard H.264, while maintaining similar video quality. It is expected to be readily adopted and deployed in	GIỚI THIỆU 

wide variety of video applications.

In video compression, video filters are the multidimensional signal processing tools used in video codecs to improve compression efficiency while keeping compression artifacts to the minimum possible levels. Such filters can be used prior and/or post to encoding. They are employed to remove blocking and compression artifacts in the encoded pictures. They are also helpful in improving visual quality and temporal prediction as filtered frames are used as reference for temporal prediction while encoding succeeding frames.

HEVC uses multiple concatenating post-filters during encoding/decoding process and are commonly referred to as loop filters, they include 1) deblocking loop filter (DBLK) and 2) Sample adaptive offset (SAO) filter[2]. The deblocking loop filter has been inherited from H.264/AVC and alleviates the blocking artifacts. Whereas SAO placed in the reconstruction loop after the deblocking process helps reduce the quantization error introduced by lossy compression of the input source. This SAO filtering is the process of adding offset to deblocked pixel value according to contexts (SAO

type) such as edge direction and shape (edge offset (EO)), and pixel level (band offset (BO)). The idea is to classify reconstructed pixels into different categories and then reduce the distortion by adding a (different) offset for each category of pixels. The offsets to be added are derived by the encoder using average reconstruction error for each of the bands and transmitted in rate-distortion optimized way while conforming to the standard specification. Since the characteristics of a picture may vary with locations, SAO divides a picture into largest coding unit (LCU)-aligned regions to obtain local statistical information and derives different offsets for each LCU.

Thus the task at hand is to design low complexity SAO encoder architecture suitable for hardware video encoders (VE). The architecture needs to be such that it can perform statistics collection, derivation of offsets for each of the pixel classes followed by rate-distortion(RD) optimized selection of the transmission parameters for encoding. This paper presents a high performance, silicon area efficient, and software configurable hardware architecture for SAO encoding. This unified architecture for luma and chroma channels is based on single LCU pipelined

VE architecture extension of the one given in [3]. It uses block based processing with three pipeline stages for statistics collection, simplified processing on tile/slice boundaries, and multiple RD engines to achieve high performance.

Further the paper is structured as follows. Section II presents the proposed SAO encoder architecture with details of overall partitioning of the functionality and data flow, statistics collection engine, RD optimization (RDO) engine and method for tile/slice boundary handling scheme. Section III provides internal memory, area requirement along with expected gate count and computational speed of the hardware. Section IV provides details of the experiments conducted to simulate qualitative performance of the architecture.

II. PROPOSED SAO ENCODER ARCHITECTURE

The simplified architectural block diagram of the SAO Encoder Block (EB) is shown as below in Fig. 2. The EB can be divided in three categories of sub-blocks, external interface and communication blocks which are shown to the left of state machine controller block (SAO-E Controller) and the blocks involved in the unified data flow path of encode process for luma and chroma channels shown on right had

side of the SAO-E Controller. This paper primarily focuses on the data path of the architecture and this section discusses architectural and functional details of the statistic collection engine and SAO parameter estimation engine in detail.

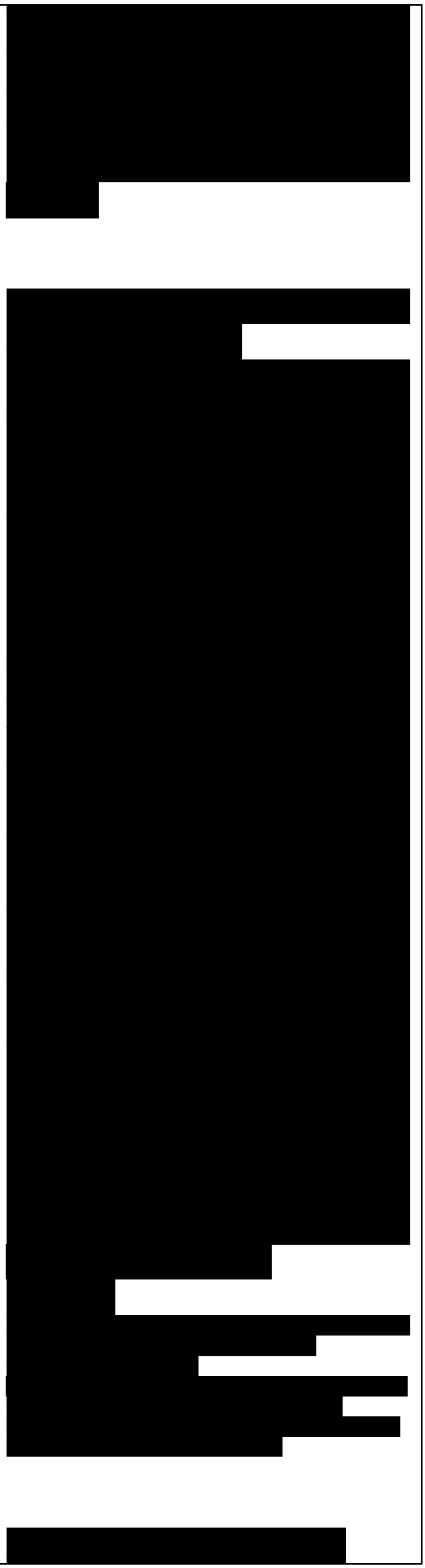
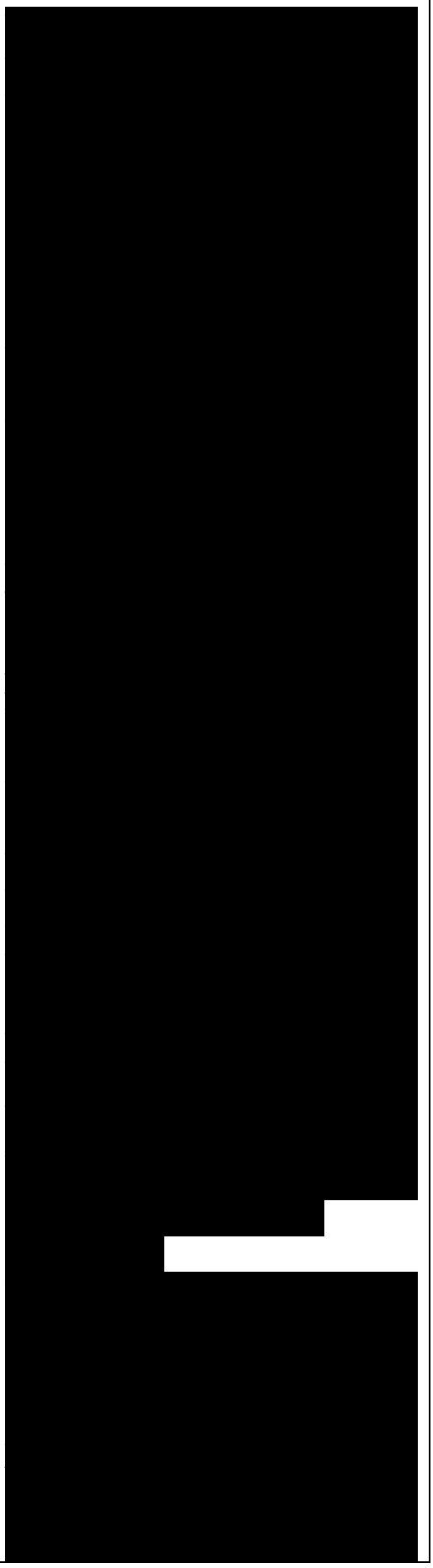


Fig. 2. Proposed SAO Encoder Architecture: Block diagram

tile instead of one pixel at a time. The column major order of sub-tile processing used in the architecture is depicted in Fig. 4, it also shows the 3x3 sub-tile block which is actually used to collect statistics for the central sub-tile. In order to utilize the adjacent sub-tiles fetched in the scratch memory and to optimize the memory accesses, the sub-tile statistics collection is internally pipelined in three stages as shown in Fig. 5. This architecture enables read/write of entire block in one cycle from local memory instead of multi-cycle access assuming traditional 1D line arrangement, improving performance. The last row and column of sub-tiles are not used in the statistics collection as these pixels are not deblocked in the LCU pipelined encoder architecture.

A. Statistics Collection Engine

The RD optimized selection of the transmission parameters require the LCU level contextual statistics of reconstruction error and number of pixels to be collected in 48 classes, of which in first group of 32 classes pixels are classified according to the pixel intensities (related to BO estimation) and in remaining 16 classes the pixels are classified into the group of four groups of four classes each according to orientation and nature of the edge at the pixel locations (related EO estimation). The process of statistics collection is shown in Fig. 3. Where it can be seen that each pixel in the LCU is classified five times once based on pixel intensity, and four times according to the edge type in four orientations. This is followed by accumulation of reconstruction error and pixel count in appropriate error and pixel count accumulators. For simplicity of hardware (HW) the proposed architecture does not validate availability of the pixels from adjacent LCUs while collecting EO specific statistics for the pixel on top or left of the LCU boundaries irrespective of the tile/slice boundary.



Further, the statistics collection is accelerated by doing statistics collection for 2D pixel blocks of size 4x4 called sub-

Fig. 4. Proposed 2D pixel block based statistics collection process

Fig. 5. Proposed 3 stage pipeline in Stat Engine

B. Parameter Estimation Engine

The second part of the EB is a parameter estimation engine which consumes the statistics collected by the statistics collection engine with SAO parameters for top and left LCUs in the picture to estimate the SAO parameters in RDO sense for current LCU. Thus the process involves selection of the best sample adaptive offset for all 48 classes of pixels, selection of SAO type, merge cost estimation and SAO parameter selection. The sub-blocks of EB that are part of the parameter estimations engine include offset RDO engine, SAO type selection engine, merge cost computation block and merge decision engine. These blocks are explained further in this section.

1) Offset RDO Engine

Offset RDO engine performs the task of distortion and RD cost computation for each class of pixels for a given candidate

offset and then selects an offset o that minimizes the RD cost J . For a class of pixels the RD cost is defined as

J = JSAO - Jorig = Do +%Ro (1)

Where JSAO is the RD cost of using SAO for the class of pixels and Jorig is the RD cost of disabling SAO for the class of pixels; Do is the reconstruction error between encoded pixels and SAOed reconstructed pixels, Ro is the bitrate of ending the offset o and X is the Lagragian multiplier used to convert the rate associated with offset into the distortion value. If the concerned class of pixels contains N number of pixels with cumulative reconstruction error of E before SAO then Do is defined as [4]

Do = N x o2 - E x o (2)

Proposed architecture uses a HW block to calculate the RD cost for all candidate offsets for a class of pixels and select the offset value that minimizes the RD cost. One such block RDO for offset #1 is shown in Fig. 6. The architecture uses 48 similar blocks to parallelize the offset RDO process. Each of these blocks takes N, E, R and X values as input and gives out the RD optimal offset o along with associated RD cost J . At this point the standard defined sign constraints on the edge offsets is validated and if found to be noncompliant the offset is set to zero with appropriate RD

cost.

2) SAO Type RD Cost Engine

For a color channel of the video sequence, the SAO parameters to be encoded into the bitstream include merge mode, SAO type and four number of offsets. The SAO type RDO engine uses the RD costs computed for 48 classes of pixels and selects the 4 consecutive classes of pixels from first 32 (BO), starting from the ith class such that

JB = min fe Jj\ (3)

1<;<28 J

The engine also computes the cumulative RD cost for the class of pixels corresponding to four edge orientations or EO types. The engine then passes this information (five RD costs and 20 offsets) to the Merge Engine which utilizes this information to finally select the SAO parameters for the current block.

Fig. 6. Proposed Offset RDO Engine

3) Merge Cost Computation Block

The merge cost computation block is again a unified HW block for both luma and chroma channels. This block uses SAO parameters from the candidate LCUs for merge and corresponding statistics collected for current LCU to estimate the merge RD cost J which is defined as

JMerge DMerge ^ XRMerge (4)

Where, DMerge is computed as

defined in (2) and RMerge is the bitrate for SAO merge signaling in the encoded bit-stream. 4) SAO Parameters Selection Engine The EB sub-blocks discussed earlier in this section are the unified blocks for both luma and two-chroma channels. Thus in the encode process these blocks process three channels sequentially one after another and collect the SAO type RD cost and offsets for each of them. Merge costs for two candidate LCU parameter sets are also computed and are passed on to the SAO parameters selection engine which selects the best set of parameters that minimize the SAO RD cost for current LCU. Architectural details of the engine are shown in Fig.7. The engine compares the RD costs of the six candidate SAO types (1-BO, 4-Eo and SAO-OFF) for luma channel and selects the best SAO type. For the chroma channels the RD costs of the six candidate SAO types for two channels are combined and then compared to choose the SAO type that minimizes the joint RD cost. After this, the joint RD cost of the selected SAO types for luma and chroma channels is compared with RD cost for valid merge candidates (based on configuration option setting for filtering across slice/tile boundaries) and the one which minimizes the RD	

cost is chosen and corresponding SAO parameters are made available for encoding in to the bit-stream and to the SAO decoder for actual SAO filtering.

Fig. 7. Proposed SAO parameters selection engine

5) External Overrides and Additional Statistics for SAO On/Off algorithms

Proposed architecture allows external overrides to disable SAO process selectively for luma or chroma channels, specific SAO types, merge candidates, restrict offset values to a range. This information can be programmed into the EB via MMR. SAO-E Controller block uses the information to control operations of the offset RDO Engine, SAO type RDO engine and multiple comparators present in the SAO parameters selection engine, make those overrides effective. Additionally the EB also collects the SAO on/off statistics at frame level such that it can be used to fine tune frame level SAO encoder algorithms for rate-control and avoidance of artifacts.

III. EXPERIMENTS

The proposed architecture has been implemented in Verilog HW description language and synthesized for core data path. The top level HW area and cycles are estimated based on fully implemented H.264 design based on similar architecture [3][5]. The design

can reach 266 MHz in 28 nm CMOS process node. It can encode Ultra HD (4K) resolution 60fps video at 200 MHz of clock. The table I provides area and cycles details, which have been estimated from core data path synthesis and top level estimation for previous designs. The overall design is estimated to take around 0.15 mm2 after final place and route with assumption of 60% utilization. The video quality simulation shows on average 3.6% bit-rate saving for Random Access (RA) configuration and 4.3% bit-rate saving for Low Delay (LD) configuration of HEVC as shown in table II.

TABLE I. SAO HARDWARE : AREA AND PERFORMANCE

TABLE II. SAO HARDWARE : VIDEO QUALITY SIMULTIONS

Gain of SAO overSAO-OFF encodingfor random access encoding Gain of SAO overSAO-OFF encodingfor low delay encoding

IV. CONCLUSION

In this paper, a high performance architecture is proposed for SAO encoding in HEVC to enable ultra HD resolution (4K) at 60 frames per second. It proposes different VLSI level architecture techniques to achieve desired goals. The proposed design is estimated to take 0.15 mm2 in 28 nm CMOS process and is

area efficient with bit-rate saving in line with HM software.	