2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)

Hardware architecture design of HEVC entropy decoding

1st Shiyu Wang
School of Computer Science, Northwestern Polytechnical University
Engineering Research Center of Embedded System Integration, Ministry of Education
Xi'an, China
onion0709@mail.nwpu.edu.cn

2nd Shengbing Zhang
School of Computer Science,Northwestern Polytechnical University
National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology,Northwestern Polytechnical University
Engineering Research Center of Embedded System Integration, Ministry of Education
Xi'an, China
zhangsb@nwpu.edu.cn

3rd Jihe Wang
School of Computer Science, Northwestern Polytechnical University
Engineering Research Center of Embedded System Integration, Ministry of Education
Xi'an, China
wangjihe@nwpu.edu.cn

4th Libo Chang
School of Computer Science, Northwestern Polytechnical University
Engineering Research Center of Embedded System Integration, Ministry of Education
Xi'an, China
changlibo@mail.nwpu.edu.cn

5th Liangyou Feng
School of Computer Science, Northwestern Polytechnical University
Engineering Research Center of Embedded System Integration, Ministry of Education
Xi'an, China
fly123@mail.nwpu.edu.cn

6th Xiaoya Fan
School of Computer Science, Northwestern Polytechnical University
Engineering Research Center of Embedded System Integration, Ministry of Education
Xi'an, China
fanxy@nwpu.edu.cn

***Abstract*—As the latest generation of digital video coding standard, HEVC has technically optimized multiple modules such as related frame prediction, block processing and entropy coding in the frequency coding and decoding framework. However, flexible and efficient coding algorithms make the amount of calculation in the video decoding and reconstruction process increase dramatically. The energy efficiency of traditional processors is limited, and the decoding calculation process is difficult to meet the current needs of ultra-high-definition video playback. For the most important and time-consuming bitstream analysis and entropy decoding part of the HEVC decoding process, a new hardware architecture strategy is provided, which can effectively improve the HEVC decoding performance. In this paper, the designed data access module, bitstream analysis and entropy decoding unit are simulated and verified. Build an FPGA verification platform, and use the main tier standard test sequence to simulate and verify the designed hardware acceleration architecture. The experimental results show that the hardware architecture of bitstream analysis and entropy decoding designed in this paper can reach the functions and performance indicators specified by the HEVC standard Level-4 main tier, and the bitstream analysis acceleration effect is good. This thesis aims at the current HEVC decoding calculation characteristics, and through the design of hardware architecture verification, it provides new solutions and hardware architecture ideas for subsequent HEVC encoding and decoding performance optimization.**

Keywords-HEVC; FPGA; bitstream; entropy decoding; video compression

I. INTRODUCTION

The vision plays an important role in the process of humans obtaining information. Scientists have done a lot of experiments, and the results show that 85% of the information that humans obtain from the outside comes from vision [1]. With the rapid improvement of the technological level, especially the advent of the mobile Internet [2][3], big data [4][5], and AI [6][7], the application value of digital images in production and life is getting higher and higher. In communication [8], embedded system [9][10], cloud systems [11], and other fields [12], people's demand for image information is far greater than text and other information. In the information age, with the continuous improvement of computer processing performance and the rapid development of network and sensor technology [13][14], people's demand for

978-0-7381-2646-3/21/$31.00 ©2021 IEEE
DOI 10.1109/ISPA-BDCloud-SocialCom-SustainCom52081.2021.00157

multimedia data applications has been increasing [15]16]. Video applications have become an indispensable part of people's lives because of their super expressive power, Mass video data traffic has already penetrated into all aspects of people's lives and work [17]. A survey shows that 90% of the total Internet traffic is video data, and video data traffic in the mobile Internet also accounts for 50% of the total traffic [18]. While the demand for images is increasing, the resolution and frame rate of images are constantly increasing. From standard definition, high definition to ultra high definition, to the newly emerging 4K video, the resolution has been increased from 720×576 to 4096×2160 respectively. The frame rate of many videos has also gone from 25 frames per second to 30 frames per second, and even 60 frames per second, 120 frames per second or higher [19].

In addition to the field of life and production, video images in the aerospace field are also becoming more and more important. With the continuous development of aerospace technology in my country and the world, video images have become more and more important in space exploration. More and more advanced aviation optical equipment is used in space technology exploration, which makes the amount of image data increase day by day. The storage space and transmission bandwidth on the spacecraft are very precious. At the same time, the captured image data is also extremely important. During compression, there should be no loss of too much quality or even zero loss. The storage and transmission of a large amount of image data has become a problem. Therefore, it is also very important to study new video compression standards for the compression of aerospace image data [20].

However, because of the huge amount of video signal information, its transmission and storage have caused a lot of trouble to people. For example, for an uncompressed video with a resolution of 1920×1080, a sampling format of 4:2:0, an effective bit width of 8bit, and a frame rate of 60fps, the amount of data contained per second can be computed as $1920\times1080\times1.5\times8\times60 \approx 1.49$Gbit. Taking a movie for one hours, storing such a movie requires 670GB of space, which will be a huge challenge for video transmission and storage.

Uncompressed video carries a lot of redundant information, such as spatial redundancy formed by intra-frame correlation, temporal redundancy formed between adjacent frames, and visual redundancy formed by the human visual system [21]. With the advent of the Internet era and the popularization of 4G, in the face of the application and rise of more and more video apps, people have begun to pursue 1K, 2K, 4K and even higher-resolution videos, and there is no way to achieve this with H.264. Required real-time codec requirements. Compression and encoding of original video frames are a necessary condition for various video applications. Certain technical means are used to eliminate redundant information, reduce the amount of data while ensuring considerable restoration quality, and reduce the burden of video transmission and storage. Therefore, whether it is in the field of professional video, consumer video, or network video applications, users' demands for high-resolution and high-compression video applications are constantly increasing. In order to meet and adapt to the growing video application needs of users, at the same time, to solve the problem of transmitting higher-resolution videos under limited bandwidth, it is necessary to further improve the efficiency of video coding and decoding, so that high-resolution videos can be stored and transmitted more effectively.

HEVC is a more efficient video coding standard, and its prediction process becomes more complicated. Compared with the previous coding standards, the HEVC intra-prediction process introduces more prediction directions and more flexible and diverse block partitioning methods based on quad-tree partitioning [22]. The main goal of the HEVC coding standard is to double the compression efficiency of high-resolution video images and reduce the bit rate of the video stream by 50% on the basis of the H.264/AVC standard, while ensuring the same video image quality, and then better adapt to a variety of different network environments, and at the same time can more easily implement parallel coding and decoding. However, this increase in compression rate has also caused the complexity of the HEVC codec algorithm to increase rapidly. Compared with H.264, the complexity of the HEVC standard has increased by almost 2 to 3 times, which also affects its performance to a certain extent. Running real-time HEVC video decoding faces huge challenges. This has prompted many researchers to work hard to improve the algorithms used in HEVC.

In order to solve the problem that the transmission bandwidth [23] and storage space [24] cannot be satisfied due to the amount of video data is increasing, it is necessary to improve the compression efficiency of the video data. The new video coding standard has greatly improved the coding performance of the video, and the coding quality has been greatly improved under the unit bit rate, but with it, the computational complexity has increased rapidly. There are two main directions of current image coding research. One is to improve the coding performance of the image, that is, to improve the coding quality of the image at the same bit rate; the other is to improve the compression speed of video data, including algorithms, under the premise of ensuring the coding quality. Aspect optimization and optimization of the hardware platform. However, due to the low throughput of the software solution [25], it cannot meet the real-time requirements, which leads to the demand for hardware accelerators. Hardware accelerators allow video encoders to meet the real-time requirements of these applications while executing computationally intensive algorithms [26][27].

For a single-core processor system, the processing speed of the video data signal can be increased by increasing the processor clock frequency, but this will consume higher power [28], and the corresponding heat dissipation problem cannot be ignored [29]. At present, the clock of the single-core processor is simply increased. The frequency and increase its power consumption [30] have reached the limit, and single-core processors can no longer meet people's needs for high-resolution video applications.

With the rapid development of multi-core processors, the HEVC codec standard has gradually got rid of hardware limitations, and multi-core processors have become the key to solving this problem [31]. A series of parallel designs on a multi-core processor and parallel computing operations on these design units through multi-core resources can effectively improve the performance of the decoder. Qiu et al. had proposed several novel approaches in scheduling for low-power high-performance multi-core systems [32][33][34]. It is worth noting that the multi-core processor does not simply superimpose multiple core resources in function. When performing video encoding and decoding operations, the entire task module needs to be designed and divided reasonably and effectively, and each can be processed in parallel [35]. The unit allocates corresponding core processing resources, and performs reasonable and effective scheduling of the core resources of the multi-core processor, and maximizes the use of the computing performance of each core resource, so that efficient parallel processing of video application software can be realized on the multi-core platform , improve the overall performance of decoder decoding.

This paper presents a hardware parallel acceleration architecture of HEVC entropy codec based on FPGA. The memory access process is reasonably organized in the calculation process, and a special buffer and state machine are designed. The scheme realizes the process of stream parsing and entropy encoding and decoding, and the architecture effectively improves the efficiency of decoding stream parsing

II. RELATED WORK

The HEVC standard was born for parallel computing and processing [36]. Due to the increasing demand for video applications, the research on high-efficiency parallel coding and decoding technologies is becoming a hot spot. However, the complexity of the HEVC standard has increased a lot compared to previous generations of standards, and it is difficult to achieve major breakthroughs in pure software code calculations, so researchers will combine multi-core hardware platforms for corresponding research. It is particularly important to design efficient parallel algorithms on multi-core hardware platforms [37][38] and effectively schedule multi-core resources [39] to solve the load balancing problem.

Many domestic scholars have done some research on video coding and decoding implementation methods on multi-core processors. Regarding the parallel decoding based on multi-core, the document divides the HEVC decoder into three task modules. The first part is the entropy decoding task module, the second part is the pixel decoding module based on CTU lines, and the third part is based on the pixel decoding module. Deblocking filter module of CTU line [40]. For the latter two task modules, a parallel algorithm based on CTU lines is designed [41]. A single core processor is used to decode the same line of CTU serially. At the same time, the CTU line dependency between the task modules is used to realize the parallel calculation and processing of the decoder. Literature studied a parallel method of deblocking filtering. In this method, the decoder allocates a balanced number of CTU rows to each thread according to the number of CTU rows in a frame of image and the number of core processors, and then each thread will Perform vertical boundary filtering operations on the multiple CTU lines it is responsible for. After the vertical boundary filtering operation of a frame of image is completed, then these threads are used to process all horizontal boundaries in the CTU line, and all the horizontal boundaries in the CTU line are processed until the current image frame is completed. After the filtering operation, all threads will repeat the above operation for the next image frame. In addition, regarding the research on parallel coding based on multi-core, literature proposed a HEVC multi-granularity fusion parallel coding method, which combines different granular units and uses the CTU row-level parallel coding algorithm based on the WPP idea at the CTU level. Parallel coding of different CTU lines in different frames uses parallel intra prediction algorithms at the CU level to implement CU parallel coding at different depths, thereby realizing a multi-granularity parallel method combining CTU level and CU level [42].

After the release of the new video coding standard, many software and hardware products supporting the H.265/HEVC standard have emerged at home and abroad. In May 2016, ARM released a new Video Processing Unit (VPU), code-named Egil. Egil fully supports H.265/HEVC encoding and decoding. Egil has a maximum of 8 cores, a single core 800MHz can handle 1080P, 120fps encoding and decoding, and 6 cores can support 4K, 120fps real-time encoding and decoding. Apple's A8 processor chip integrates a video encoder and decoder that supports the H.265/HEVC standard, and can perform real-time video encoding and decoding [43]. Starting from A10, H.265/HEVC 10bit video codec has been supported. Its official claim can achieve 60fps H.265/HEVC video codec. As a hardware video encoder, Fudan University's H.265 Video Encoder IP Core implements most of the functions of H.265/HEVC. It supports the Main grade of HEVC/H.265 and the image input of YUV420p [44]. The image types include I image and P image. It is implemented on the FPGA platform and can perform real-time encoding up to 4K and 30fps. In 2018, Peking University released a new video encoder xAVS (V1.0), which complies with the latest domestic video encoding standard AVS2, which can realize real-time encoding of high-definition video images. In the slow grade, its encoding speed is faster than x265.

In general real-time video image encoding processing, thread-level parallel processing is achieved by dividing tasks or data in video encoding applications. Because the digital signal processor DSP can perform complex mathematical operations while ensuring real-time processing and has good programmability, it has become a good platform for implementing video H.264 and HEVC encoding [45]. With the development of semiconductor industry technology, the process technology of chips continues to decline. Single-chip DSP has evolved to multi-core DSP, integrated with other functional units (such as on-chip memory and functional I/O), and further evolved to

multi-core processor system-on-chip (Multiprocessor System-on-Chip, MPSoC). Based on the ITRS roadmap, the number of cores of the on-chip multi-core processing system will exceed one thousand in 2022. However, with the increase in the number of cores, the chip area has increased sharply, the cost increase and the increase in power consumption caused by the production process problems are unavoidable. In addition, the long interconnection lines between the cores will cause large delays. These problems are further increased. Improving chip performance brings challenges.

Since the intra-frame and inter-frame encoding algorithms take more than 90% of the total encoding time, the optimization and acceleration of this part of the algorithm can greatly improve the video encoding efficiency. However, the current research is mostly based on general-purpose processor platform development, which is not suitable for hardware implementation. For example, the existing intra-frame optimization algorithm that relies on spatio-temporal correlation increases the bus load of the hardware circuit due to the need to extract adjacent frames or adjacent coded block information. According to the image texture detection to eliminate part of the CU tree segmentation unit or the image frame texture mode, this type of algorithm is more hardware friendly than the spatio-temporal correlation algorithm, but the existing texture detection algorithm is too complicated, such as Sobel operator extraction, pixel probability distribution statistics, etc. The hardware implementation cost is relatively high, so it is necessary to develop a more suitable hardware processor unit to solve the above problems.

At present, most of the research on video image coding processing focuses on the module function research of the H.265 algorithm, and the research on parallel processing mainly focuses on the software level. Aiming at the most important and time-consuming bitstream analysis and entropy decoding part of the HEVC decoding process, the hardware acceleration structure and parallel design methods are developed, and new solutions and hardware architectures are provided in terms of improving computing speed and parallel computing capabilities. It is very important to improve the entire HEVC decoding performance. This paper proposes an FPGA-based HEVC bitstream analysis and entropy decoding hardware parallel acceleration architecture, and reasonably organizes the memory access process in the calculation, and designs a dedicated buffer and state control unit in the architecture. The results show that this architecture effectively improves the parsing efficiency of the bitstream and the entropy decoding process.

III. ENTROPY CODING AND DECODING OPERATION

Aiming at the most important and time-consuming bitstream analysis and entropy decoding part of the HEVC decoding process, the hardware acceleration structure and parallel design methods are developed, and new solutions and hardware architectures are provided in terms of improving computing speed and parallel computing capabilities. It is very important to improve the entire HEVC decoding performance. Fig.1 shows the overall flow of HEVC encoding and decoding.

CABAC entropy coding and decoding is an optimization of traditional arithmetic coding and decoding [46]. First, the CABAC entropy encoding and decoding process uses integer operations; secondly, the CABAC entropy encoding and decoding process uses two-system operations; the third CABAC entropy encoding and decoding process uses a look-up table calculation method for multiplication calculation.

Unlike variable length coding, traditional arithmetic coding is a coding and decoding method based on recursive partitioning. Its essence is to assign a codeword to the entire input sequence, instead of assigning a codeword to each character in each input sequence. On average, a single character can be assigned a code with a code length of less than 1. Therefore, traditional arithmetic coding can get close to optimal coding results. According to the probability of different symbol sequences of the source, the interval [0, l] is divided into non-overlapping sub-intervals, the width of which is the probability of each sequence symbol, so that each sub-interval corresponds to the source symbol sequence one-to-one. In short, the corresponding symbol sequence can be represented by any real number in the subinterval, and this real number is the codeword corresponding to the symbol sequence.

Compared with the traditional arithmetic decoding, the arithmetic decoding in CABAC has three main improvements. One is to optimize the limitation of calculation accuracy. Since integer calculations are much simpler than floating-point numbers, for a video sequence with a bit depth of 8, the coding interval is [0.5 10]. The second is to change other hexadecimal code sequence symbols to binary code sequence symbols, so the source symbols are only 0 and 1, and the symbol probability has only a large probability value (PLPS>0.5) and a small probability value (PLPS< 0. 5). This improvement gives the CABAC decoder a binarization module, which converts non-binary values into corresponding binary strings.

The third is to change the multiplication operation of the probability interval update to the table look-up calculation. In CABAC, the probability of the source symbol is uncertain. In order to avoid multiplication, discretization interval threat > and probability estimation of the context model, the multiplication operation is changed to look-up table calculation, and the low probability symbol LPS (PLPs<0.5) is always estimated), the probability is discretized into a state table of 64 states, and the probability is calculated by looking up the table by the index of the current character to be coded in the state table. This improvement enables the CABAC decoder to have a context modeling module, which selects appropriate decoded syntax elements to build a model to estimate the conditional probability of the current decoded element.

Entropy encoder is the last stage of video encoder [47]. The input of the entropy encoder includes quantization coefficients, intra prediction mode, motion vector, block size, sample adaptive offset (SAO) parameters and other control flags. First, these data are serialized to form syntax

element streams (SEs). After serialization, the entropy encoder performs binarization. The result is to generate BIN according to the encoding rules dedicated to each SE [48].

BIN has different statistical distributions. They are coded with equal or variable probabilities of two values. In the first case, the bypass mode (BM) is used, and in the second case, adaptive PMs are used to determine the probability. The BIN obtained for some SEs can only include one or two of the two groups. The mode assignment depends on the position of the SE in the string of boxes.

For a variable probability coded box, the context is used to select the corresponding PM. The context label is related to the type of SE, the bin position in the binary representation, the 2-D neighborhood, and the value of the same type SE before the current type. Each PM includes the Most Probable Symbol (MPS) value and a 6-bit index corresponding to the probability of the Least Probable Symbol (LPS). The input bin is compared with the MPS value to determine whether to encode LPS or MPS. The selected PM is directed to the BAC and updated after encoding in the BAC. Please note that the pm pointing to BAC contains a 1-bit symbol (LPS/MPS), not bin. During the update/adaptation process, the indexes in the selected PM of MPS and LPS are incremented and decremented, respectively. Therefore, a higher value corresponds to a stronger probability asymmetry. If the index is 0 and LPS is encoded, the MPS value is inverted.

Use adaptive PMs to encode the importance map and logo library. The context label used to select pm consists of offset and increment. For the saliency map, the increment depends on the position of the corresponding coefficient. Taking into account the non-zero contribution in the previous/adjacent sub-blocks, one of the four context increment maps is selected. The context offset is determined by the block type (luminance/chroma, DC/AC, N value).

Some coefficient data written into the bitstream is coded by BM, and BM assumes that the probability of the two bin values is equal. Such data includes the signs of non-zero coefficients and the remaining absolute values of coefficients that are not fully described by the GT1 and GT2 flags. These values are obtained as the difference between the actual coefficient level and the basis (1, 2 or 3) determined by the GT1/GT2 logo. These values are coded with Golomb-Rice codes, and the conditional increment of the Rice parameter is between 0 and 4. When the encoded coefficient amplitude exceeds the continuous threshold (3, 6, 12, and 24), there will be an increase.

According to statistics, CABAC occupies more than 25% of the time overhead in the overall HEVC encoding and decoding process, and the data generation rate of residual transform coefficients accounts for 60% to 86% of the total syntax element generation rate. In recent years, a large number of researches on CABAC entropy decoders have been conducted at home and abroad, and excellent results have also been achieved, which has continuously improved the speed of CABAC entropy decoders. CABAC entropy decoding is mainly used to analyze various control information, logo information, motion information, and residual information used in image reconstruction. It is one of the most important algorithms and steps in the HEVC standard. Its performance is important to the entire decoding process. Have an important impact.

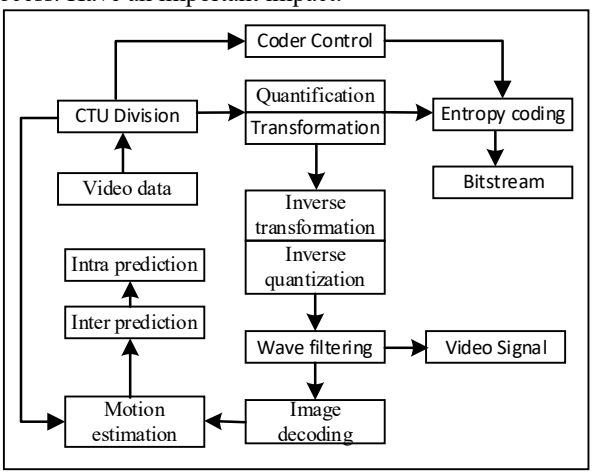

Figure 1. Overview of the Hevc encoding and decoding process

IV. ARCHITECTURE DESIGN OF ENTROPY DECODER

This paper designs a complete entropy decoder. The overall architecture includes Bitstream parsing Finite State Machine (B-FSM), residual coefficient fast scanning module, initialization module, modeling module, bitstream buffer, arithmetic decoding unit and non-binarization area, as shown in Fig. 2.

The bitstream analysis control module sequentially determines which syntax elements exist in the current CTU according to the bitstream sequence, and controls the coordination of other modules. Because the control logic of the residual coefficient scanning is more complicated, the residual coefficient fast scanning module is added to quickly scan the residual coefficient, so that the state is directly transferred from the position information decoding to the amplitude information decoding state.

The modeling module includes a context index generation unit and a context storage area, and predicts the probability of the next syntax element to be decoded according to the decoded word. The context index generating unit calculates the context index of the current decoding syntax according to the control signal of the bitstream parsing state machine controller and the context index increment algorithm in HEVC, and outputs it to the context storage area. The context storage area temporarily stores the initialized context variables, and subsequently reads the context variables in the storage unit according to the index, and outputs them to the arithmetic decoding unit.

The bit stream buffer area is used to provide test bit stream data for saturation of the arithmetic decoding unit, and extract the required dynamic length bits according to the requirements of the arithmetic decoding unit.

The arithmetic decoding unit includes a normal decoding area, a bypass decoding area, and a termination decoding area. The regular arithmetic decoding area is used to decode modeling syntax elements. First, initialize the

length and offset of the arithmetic decoding space interval; obtain the probability interval value of the current decoded syntax element by looking up the LUT method, and calculate other variables to decode to obtain the decoded binary value. The second step is to update the variable and write it back to the context storage area; finally, initialize the arithmetic decoding space to prepare for the next operation.

The non-binarization area judges whether the current decoding syntax is fully decoded according to the decoded binary value of the arithmetic decoding unit, the bitstream control unit instruction and the non-binarization process. After the decoding is completed, the binary value is calculated into the corresponding data of the syntax element and provided to the video decoding unit.

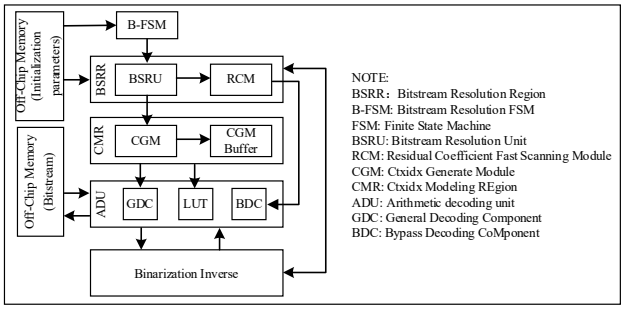

Figure 2. Hardware architecture for entropy decoding

V. USING THE TEMPLATE

The simulation verification FPGA development board uses V6-XC6VLX760. Then integrate the various modules into the CABAC top-level module, extract the standard bitstream test sequence, and perform functional simulation and FPGA verification on the entire CABAC decoder. The bitstream analysis, entropy codec module and data access module are verified in FPGA. Table I lists the logic resource occupation of FPGA.

TABLE I. FPGA RESOURCE OCCUPATION

Parameter	Value
Number of slice registers	6473
Number of LUTs	8404
Number of block RAMs	69

Parameter quantization usually determines the encoding compression rate. The smaller the quantization value, the lower the actual compression ratio. Facing the same data, the code rate is higher, and the amount of decoding syntax elements is larger. And the smaller the quantization parameter value, the lower the loss rate of image coding, and the finer the image output result. Fig. 3, The picture on the right is a partial image of the first frame decoded when the quantization parameter is 22 for the KristenAndSara 1080P bitstream, the picture on the left is for the decoding situation when the image quantization parameter is 37. Comparing the two pictures, it can be clearly seen that when the quantization parameter is 22, the image is more delicate.

Table II records that the extraction and decoding modules designed in this thesis can decode the bit rate of 2,256.8 KB/s, and the bitstream compression rate of 35.1. The main index of Level 4 in the HEVC standard has a bit rate standard of 1,500 KB/s, and the bit rate compression rate is greater than 4; Obviously this design meets the index requirements of the main index in Level 4.

By comparing the generation rate of syntax elements at each level in Table II, it can be seen that the rate of generation of residual data can reached 85% of the total generation rate of syntax elements. Optimize storage and reading, which can further improve the data transmission capacity between each module; Similarly, targeted optimization of the decoding of syntax elements can improve the overall decoding performance more effectively. It can also be seen from the table that the video with the same content and the same resolution has different quantization parameters, and the coded stream rate is also very different: the larger the quantization parameter value, the smaller the interval obtained after quantization, the more serious the loss of details, and the lower the data rate of each syntax element when sending samples; the smaller the quantization parameter, the more quantized intervals, the richer the details of the image, and the higher the data rate of each syntax element.

Table III shows the cycle occupancy state when the standard bitstream through hardware decoding (quantization parameter: 22; clock frequency: 400MHz). As shown in Table III, Other analysis modules accounted for the most of the cycle, concentrated in about 40%. The average cycle of entropy decoding is 36%. Columbus decoding cycle is the least, less than 0.01%. Therefore, in the subsequent optimization scheme, it is necessary to focus on how to increase the prediction range during bitstream parsing, so as to further increase the decoding rate of HEVC.

(a) Quantization parameters：37

(b) Quantization parameters：22

Figure 3. Image reconstruction with different quantization parameters

TABLE II. STANDARD BITSTREAM SIMULATION(DATA RATE)

	Basketball DrillText 832*480		BQSquare 416*240		ChinaSpeed 1024*768		BQTerrace 1920*1080	
QP	37	22	37	22	37	22	37	22
RD(KB/s)	16.45	181.77	7.85	101.58	72.33	461.51	76.4	2388.21
SAO(KB/s)	0.71	22.37	0.12	0.36	0.97	3.52	1.82	7.67
PU(KB/s)	3.03	12.21	1.4	4.33	7.76	28.96	7.9	70.775
CU(KB/s)	9.61	33.68	3.43	8.01	26.4	73.9	27.17	118.4
MVD (KB/s)	2.65	11.48	0.82	2.76	7.28	30.6	4.13	42.18
Other(KB/s)	6.86	24.05	2.27	9.58	18.9	56.4	20.36	159.66
Total(KB/s)	39.33	285.58	15.91	126.65	133.66	654.9	137.8	2,786.913

NOTE: QP: Quantitative parameters; ;RD: Residual data

TABLE III. STANDARD BITSTREAM SIMULATION(OTHER)

	Basketball DrillText 832*480		BQSquare 416*240		ChinaSpeed 1024*768		BQTerrace 1920*1080	
QP	37	22	37	22	37	22	37	22
CO-R	473.2	72.9	269.7	41.2	292.3	49.8	849.7	35.1
CR(KB/s)	38.9	264.65	16.16	116.45	122.45	676.65	108.4	2,256.8

NOTE: QP: Quantitative parameters; CO-R: Compression ratio; CR: Code Rates; RD: Residual data

TABLE IV. HARDWARE DECODING SIMULATION

Circles	BasketballDrillText 832*480		BQSquare 416*240		ChinaSpeed 1024*768		BQTerrace 1920*1080	
CA-D	14932546	26.99%	7731683	46.77%	39104754	32.59%	169347452	41.64%
CD	586	0.00105%	588	0.00356%	591	0.000493%	557	0.000136%
LSCS	18353692	33.17%	803652	4.86%	36925845	30.77%	27057426	6.65%
OR	22037462	39.83%	7993585	48.36%	43953763	36.63%	210247403	51.70%
Total	55324286		16529508		119984953		406652838	

NOTE:CD: Columbus decode; LSCS: Last Significant Coeff Scan; OR: Other Resolution; CA-D: CABAC decode

VI. CONCLUSION

This paper introduces the simulation process of HEVC stream analysis and entropy decoding unit. The main index is used for most video applications. Therefore, we should carry out relevant simulations on several standard video sequences of main tier, and obtain the bit rate, compression rate, and decoding rate of syntax elements of each video sequence. The bitstream analysis and entropy decoding unit is optimized, and the module is implemented on FPGA to obtain the FPGA resource occupancy data of the module. This chapter uses multiple standard test bitstreams to verify entropy from three aspects of function, performance and resource consumption. The decoding performance of the decoding unit has been verified by FPGA simulation, and the experimental results have been analyzed. Analyzing the result data, the bitstream parsing and entropy decoding unit of this architecture scheme can decode the correct HEVC syntax elements. This article improves the structure of the state machine to reduce the cycle consumption of the bitstream parsing process; it is fast by adding residual coefficients The scanning module quickly scans the residual coefficients, predicts the next syntax element that needs to be decoded, and reduces the proportion of the scanning period of the two-dimensional position coordinate in the total period by 52.46% at the highest and 16.17% at the lowest; effectively reducing the overall The consumption period of entropy decoding. The decoding performance meets the bit rate and compression rate requirements of the main level of the HEVC standard Level-4.

REFERENCES

[1] D. Grois *et al*., "Performance Comparison of Emerging EVC and VVC Video Coding Standards with HEVC and AV1," in SMPTE Motion Imaging Journal, 2021 130(4), pp. 1-12.

[2] M. Qiu, K. Zhang, M. Huang, Usability in mobile interface browsing, Web Intelligence and Agent Systems Journal, 4(1), pp. 43-59, 2006.

[3] L. Chen, Y. Duan, M. Qiu, J. Xiong, K. Gai, Adaptive resource allocation optimization in heterogeneous mobile cloud systems, IEEE 2nd Int'l Conf. on Cyber Security and Cloud Comp. (CSCloud), 2015

[4] L. Qiu, K. Gai, M. Qiu, Optimal big data sharing approach for tele-health in cloud computing, IEEE SmartCloud, pp.184-189, 2016.

[5] Z. Lu, N. Wang, J. Wu, M. Qiu, IoTDeM: An IoT Big Data-oriented MapReduce performance prediction extended model in multiple edge clouds, Journal of Parallel and Dis. Comp., Vol.118, pp. 316-327, 2018.

[6] K. Gai and M. Qiu, Reinforcement learning-based content-centric services in mobile sensing, IEEE Network, 32(4), pp.34-39, 2018.

[7] K. Gai and M. Qiu, Optimal resource allocation using reinforcement learning for IoT content-centric services, Applied Soft Computing, Vol.70, pp.12-21, 2018.

[8] M. Qiu, C. Xue, Z. Shao, Q. Zhuge, M. Liu, E. H.-M. Sha, Efficent algorithm of energy minimization for heterogeneous wireless sensor network, IEEE Int'l Conf. on Embedded and Ubiquitous Computing (EUC), pp. 25-34, 2006.

[9] M. Qiu and J. Li, Real-time embedded systems: optimization, synthesis, and networking, CRC Press.

[10] K. Gai, M. Qiu, H. Zhao, M. Liu, Energy-aware optimal task assignment for mobile heterogeneous embedded systems in cloud computing, IEEE 3rd int'l conf. on CSCloud, 2016.

[11] M. Qiu, Z. Ming, J. Wang, L.T. Yang, Y. Xiang, Enabling cloud computing in emergency management systems, IEEE Cloud Computing, 1(4), pp.60-67, 2014.

[12] M. Qiu, E. Khisamutdinov, Z. Zhao, C. Pan, J.W. Choi, N.B. Leontis, P. Guo, RNA nanotechnology for computer design and in vivo computation, Philosophical Transactions of the Royal Society A, 2013.

[13] Wang S, Zhang S, Huang X, et al. A Highly Efficient Heterogeneous Processor for SAR Imaging[J]. Sensors, 2019, 19(15):3409.

[14] Q. Zhang, T. Huang, Y. Zhu, M. Qiu, A case study of sensor data collection and analysis in smart city: provenance in smart food supply chain, Int'l J. of Distributed Sensor Networks, 9(11), 382132, 2013.

[15] Yuan H, Wang Q, Liu Q, et al. Hybrid Distortion-Based Rate-Distortion Optimization and Rate Control for H.265/HEVC[J]. IEEE Transactions on Consumer Electronics, 2021, PP(99):1-1.

[16] M. Qiu, D. Cao, H. Su, K. Gai, Data transfer minimization for financial derivative pricing using Monte Carlo simulation with GPU in 5G, IEEE Int'l Journal of Communication Systems, 29(16), pp. 2364-2374, 2016.

[17] M. Liu, S. Zhang, Z. Fan, M. Qiu, H Infinite State Estimation for Discrete-Time Chaotic Systems Based on a Unified Model, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), Vol. 42, 2012.

[18] Salva-Garcia P , Alcaraz-Calero J M , Wang Q , et al. Scalable Virtual Network Video-Optimizer for Adaptive Real-Time Video Transmission in 5G Networks[J]. IEEE Transactions on Network and Service Management, 2020, 17(2):1068-1081.

[19] M. Cheon and J. Lee, "Subjective and Objective Quality Assessment of Compressed 4K UHD Videos for Immersive Experience," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 28, no. 7, pp. 1467-1480.

[20] Shea E, George A . OPIR video preprocessing and compression for on-board aerospace computing, IEEE National Aerospace and Electronics Conference (NAECON). IEEE, 2018. pp. 142-148.

[21] Sugito Y , Iwasaki S , Chida K , et al. A Study on the Required Video Bit-rate for 8K 120-Hz HEVC/H.265 Temporal Scalable Coding[C]// 2018 Picture Coding Symposium (PCS). 2018, pp. 106-110.

[22] W. Hamidouche, M. Raulet and O. Déforges, "4K Real-Time and Parallel Software Video Decoder for Multilayer HEVC Extensions," in IEEE Transactions on Circuits and Systems for Video Technology, 2016, vol. 26, no. 1, pp. 169-180.

[23] J. Niu, Y. Gao, M. Qiu, Z. Ming, Selecting proper wireless network interfaces for user experience enhancement with guaranteed probability, J. of Parallel and Dis. Comp., 72(12), pp. 1565-1575, 2012.

[24] Y. Guo, Q. Zhuge, J. Hu, J. Yi, M. Qiu, E. H.-M. Sha, Data placement and duplication for embedded multicore systems with scratch pad memory, IEEE Transactions on Computer-Aided Design of Integrated Circuits, 2013.

[25] B. Martin, W. Hamidouche, J. Le Feuvre and M. Raulet, "Unified real time software decoder for HEVC extensions," IEEE International Conference on Image Processing (ICIP), 2014, pp. 2186-2188.

[26] Y. Xu, K. Li, T.T. Khac, M. Qiu, A multiple priority queueing genetic algorithm for task scheduling on heterogeneous computing systems, IEEE 14th Int'l Conf. on High Performance Computing (HPCC), 2012.

[27] M. Qiu, H. Li, E.H.-M. Sha, Heterogeneous real-time embedded software optimization considering hardware platform, ACM symposium on Applied Computing, pp. 1637-1641, 2009.

[28] M. Qiu, J. Liu, J. Li, Z. Fei, Z. Ming, E.H.-M. Sha, A novel energy-aware fault tolerance mechanism for wireless sensor networks, IEEE/ACM Int'l Conf. on Green Computing and Comm., 2011.

[29] M. Qiu, Z. Chen, M. Liu, Low-power low-latency data allocation for hybrid scratch-pad memory, IEEE Embedded Systems Letters, 6(4), 69-72, 2014.

[30] Y. Gao, S. Iqbal, P. Zhang, M. Qiu, Performance and power analysis of high-density multi-GPGPU architectures: A preliminary case study, IEEE 17th Int'l Conf. on High Performance Computing (HPCC), 2015.

[31] W. Liu *et al*., "Fine-Grained Task-Level Parallel and Low Power H.264 Decoding in Multi-Core Systems," IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS), 2018, pp. 307-314.

[32] M. Qiu, J.W. Niu, L.T. Yang, X. Qin, S. Zhang, B. Wang, Energy-aware loop parallelism maximization for multi-core DSP architectures, IEEE/ACM Int'l Conf. on Green Computing and Comm., 2010.

[33] Y. Guo, Q. Zhuge, J. Hu, M. Qiu, E.H.-M. Sha, Optimal data allocation for scratch-pad memory on embedded multi-core systems, IEEE Int'l Conference on Parallel Processing (ICPP), pp.464-471, 2011.

[34] Z. Shao, M. Wang, Y. Chen, C. Xue, M. Qiu, L.T. Yang, E.H.-M. Sha, Real-time dynamic voltage loop scheduling for multi-core embedded systems, IEEE Transactions on Circuits and Systems II, 54 (5), 445-449, 2007.

[35] W. Chen, Q. He, S. Li, B. Xiao, M. Chen and Z. Chai, "Parallel Implementation of H.265 Intra-Frame Coding Based on FPGA Heterogeneous Platform," IEEE HPCC, 2020, pp. 736-743.

[36] J. Niu, C. Liu, Y. Gao, M. Qiu, Energy efficient task assignment with guaranteed probability satisfying timing constraints for embedded systems, IEEE Transactions on Parallel and Distributed Systems, 25(8), 2043-2052, 2013.

[37] L. Zhang, M. Qiu, W.C. Tseng, E.H.-M. Sha, Variable partitioning and scheduling for MPSoC with virtually shared scratch pad memory, Journal of Signal Processing Systems, Vol.58(2), pp. 247-265, 2010.

[38] H. Zhao, M. Chen, M. Qiu, K. Gai, M. Liu, A novel pre-cache schema for high performance Android system, Future Generation Computer Systems, Vol. 56, pp. 766-772, 2016.

[39] M. Qiu, C. Xue, Z. Shao, E.H.-M. Sha, Energy minimization with soft real-time and DVS for uniprocessor and multiprocessor embedded systems, ACM/IEEE DATE, pp. 1-6, 2007.

[40] Z. Feng, P. Liu, K. Jia and K. Duan, "HEVC Fast Intra Coding Based CTU Depth Range Prediction," IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), 2018, pp. 551-555.

[41] P. Xu, K. Chen, J. Sun, X. Ji and Z. Guo, "An adaptive intra-frame parallel method based on complexity estimation for HEVC," Visual Communications and Image Processing (VCIP), 2016, pp. 1-4.

[42] J. M. Ha, J. H. Bae and M. H. Sunwoo, "Texture-based fast CU size decision algorithm for HEVC intra coding," IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), 2016, pp. 702-705.

[43] Ibrahim A , Hager P A , Bartolini A , et al. Efficient Sample Delay Calculation for 2-D and 3-D Ultrasound Imaging. IEEE Transactions on Biomedical Circuits & Systems, 2017, 11(4):815-831.

[44] Zhao W , Shen L , Cao Z , et al. Texture and Correlation Based Fast Intra Prediction Algorithm for HEVC[J]. Advances on Digital Television and Wireless Multimedia Communications, 2012.

[45] H. Jiang, R. Fan, Y. Zhang, G. Wang and Z. Li, "Highly Paralleled Low-Cost Embedded HEVC Video Encoder on TI KeyStone Multicore DSP," IEEE Transactions on Circuits and Systems for Video Technology, 2019, (4)29, pp. 1163-1178.

[46] F. L. L. Ramos, B. Zatt, M. Porto and S. Bampi, "Energy-Throughput Configurable Design for Video Processing Binary Arithmetic Encoder," in IEEE Transactions on Circuits and Systems for Video Technology, 2021, 3(31), pp. 1163-1177.

[47] G. Pastuszak, "Multisymbol Architecture of the Entropy Coder for H.265/HEVC Video Encoders," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2020, 12(28), pp. 2573-2583.

[48] A. V. P. Saggiorato, F. L. L. Ramos, B. Zatt, M. Porto and S. Bampi, "HEVC Residual Syntax Elements Generation Architecture for High-Throughput CABAC Design," 25th IEEE International Conference on Electronics, Circuits and Systems (ICECS), 2018, pp. 193-196.