Regular Article

An Overview of H.264 Hardware Encoder Architectures Including Low-Power Features

Ngoc-Mai Nguyen[1,2], **Duy-Hieu Bui**[2], **Nam-Khanh Dang**[2], **Edith Beigne**[1], **Suzanne Lesecq**[1], **Pascal Vivet**[1], **Xuan-Tu Tran**[2]

[1] CEA, LETI, MINATEC Campus, F-38054 Grenoble, France
[2] SIS Laboratory, VNU University of Engineering and Technology, Cau Giay, Hanoi, Vietnam

Correspondence: Xuan-Tu Tran, tutx@vnu.edu.vn

Manuscript communication: received 29 April 2014, accepted 30 June 2014

Abstract– **H.264 is the most popular video coding standard with high potent coding performance. For its efficiency, the H.264 is expected to encode real-time and/or high-definition video. However, the H.264 standard also requires highly complex and long lasting computation. To overcome these difficulties, many efforts have been deployed to increase encoding speed. Besides, with the revolution of portable devices, multimedia chips for mobile environments are more and more developed. Thus, power-oriented design for H.264 video encoders is currently a tremendous challenge. This paper discusses these trends and presents an overview of the state of the art on power features for different H.264 hardware encoding architectures. We also propose the VENGME's design, a particular hardware architecture of H.264 encoder that enables applying low-power techniques and developing power-aware ability. This low power encoder is a four-stage architecture with memory access reduction, in which, each module has been optimized. The actual total power consumption, estimated at Register-Transfer-Level (RTL), is only** 19.1 **mW.**

Keywords– **H.264 encoder, hardware architecture, low power.**

1 Introduction

As the most popular and efficient video compression standard, the H.264 Advanced Video Coding (H.264/AVC) provides better video quality at a lower bit-rate than previous standards [1]. The standard is recommended by the Joint Video Team (JVT) formed by the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG). It contains a rich set of video coding tools to support a variety of applications ranging from mobile services, video conferencing, digital broadcast to IPTV, HDTV and digital storage media. Compared with the previous standards such as MPEG-4 [2], H.263 [3], and MPEG-2 [4], the H.264/AVC can achieve 39 %, 49 %, and 64 % of bit-rate reduction respectively [5]. However, because many coding tools have been adopted it makes the standard more complex and increases the computational time. It is very hard for software based implementation of the H.264 encoders to meet the real-time requirements of applications, especially for high-definition video (for example, up to 1080p: the HDTV high-definition video with 1080-line frames and progressive scan). Therefore, parallel processing solutions such as DSP-based, stream processor-based, multi-core systems or dedicated VLSI hardware architectures must be addressed to respond to this demand. In particular, designing Large-Scale Integration (LSI) like H.264 video encoding systems is a recent design trend in implementing multimedia systems aimed at high-throughput design for high-definition (HD) video [6–8] and low power design for portable video [9]. Indeed, the main issue is to lower power consumption for intended applications such as video transmission and play back on mobile terminals, to support real-time video encoding/decoding on battery-powered devices and, obviously, programmable processors or DSP-based implementations which cannot meet this requirement. For example, the design in [10] uses a 130 MHz ARM996 processor and it is only capable of QCIF decoding at 7.5 fps. Even if some software solutions can achieve QCIF at 30 fps, the power consumption is relatively large and may not be suitable for handheld applications. Thus, dedicated VLSI hardware encoding/decoding architectures targeting low power consumption are mandatory.

This paper surveys the state of the art on dedicated hardware implementation of H.264 encoders. Three different groups of H.264 video encoding architectures are introduced and analyzed. Classical architecture naturally cuts the encoding path into a pipeline of three or four stages. The pipelining schedule may be more balanced to avoid bottleneck or less balanced so that low-power techniques can be applied. More specific architectures were implemented to highly improve coding speed or scalability. However, these architectures are costly in terms of silicon area and power consumption. In this paper, the discussion on the state of the art mostly focuses on power features and specific low-power solutions. The paper proposes a novel architecture of H.264 video encoder, the VENGME design, where several techniques can be implemented to reduce

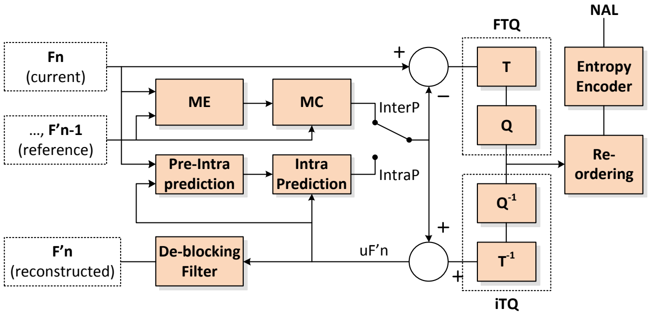

Figure 1. Functional diagram of the H.264/AVC encoder.

the power consumption. The following sections present the basic concepts of H.264 video encoding and its hardware implementations (Section 2); the state of the art on power features (Section 3) and the VENGME H.264 encoder (Section 4). Conclusions and future works will be provided in Section 5.

2 H.264 Video Encoding and Hardware Implementation

This section provides a short overview of H.264 video coding concepts. Then, the main trends of hardware (HW) encoder implementation are given in terms of power and speed requirements.

2.1 Fundamental Concepts of Video Coding

The general architecture of the H.264/AVC encoder, composed of different functional blocks, is depicted in Figure 1.

The encoding path consists of Intra prediction, Inter prediction containing Motion Estimation (ME) and Motion Compensation (MC), Forward Transform and Quantization (FTQ), Re-ordering, and Entropy encode. Intra prediction predicts the current macroblock (MB, a block of 16 × 16 pixels) based on the previously encoded pixels in the current frame, to remove spatial redundancies of video data. On the other hand, to remove temporal redundancies of video data, the inter prediction estimates the motions of the current MB based on the previously encoded pixels in different frame(s). Residual data, the differences between original current MB and predicted one, are transformed and quantized. Lastly, post-quantization coefficients are then re-ordered and entropy encoded to remove statistical redundancies. The encoded video might be encapsulated into Network Abstraction Layer (NAL) units. A decoding path that contains Inverse Transform and de-Quantization (iTQ) and Blocking filter is also built in the video encoder to generate reference data for prediction. Intra prediction uses directly the data from iTQ, while inter prediction refers to reconstructed frames from blocking filter.

In order to achieve high compression ratio, the H.264/AVC standard has adopted several advances in coding technology to remove spatial and temporal redundancies. These prominent techniques are depicted thereafter:

- A new way to handle the quantized transform coefficients has been proposed for trading-off between compression performance and video quality to meet the applications requirements. Besides that, an efficient method called Context-Adaptive Variable Length Coding (CAVLC) is also used to encode residual data. In this coding technique, VLC tables are switched according to the already transmitted syntax elements. Since these VLC tables are specifically designed to match the corresponding image statistic, the entropy coding performance is impressively improved in comparison to schemes using only a single VLC table [11];
- The H.264/AVC adopts variable block size motion prediction to provide more flexibility. The intra prediction can be applied either on 4 × 4 blocks individually or on entire 16 × 16 macroblocks MBs. Nine different prediction modes exist for a 4 × 4 block while four modes are defined for a 16 × 16 block. After taking the comparisons among the cost functions of all possible modes, the best mode having the lowest cost is selected. The inter-prediction is based on a tree-structure where the motion vector and prediction can adopt various block sizes and partitions ranging from 16 × 16 MBs to 4 × 4 blocks. To identify these prediction modes, motion vectors, and partitions, the H.264/AVC specifies a very complex algorithm to derive them from their neighbors;
- The forward transform/inverse transform also operates on blocks of 4 × 4 pixels to match the smallest block size. The transform is still Discrete Cosine Transform (DCT) but with some fundamental differences compared to those in previous standards [12]. In [13], the transform unit is composed of both DCT and Walsh Hadamard transforms for all prediction processes;
- The in-loop deblocking filter in the H.264/AVC depends on the so-called Boundary Strength (BS) parameters to deter-mine whether the current block edge should be filtered. The derivation of the BS is highly adaptive because it relies on the modes and coding conditions of the adjacent blocks.

2.2 Trends to Implement Hardware H.264 Encoder

The H.264 standard, with many efficient coding tools and newly added features, can save approximately 50 % of bit rate in comparison with prior standards [11]. Since the computational complexity of the new coding tools is very high, it is hard to implement an H.264 encoder in sequential software, especially for real-time applications [8, 14]. Two alternative solutions are multi-core software implementation and HW implementation. Both of them enable parallel computing to reduce the processing time. However, some coding tools of the H.264 are more efficiently implemented in HW. For example, most of the calculation operations in the transform process are add and shift ones. Hence, HW implementation seems to be the relevant choice. As hardware implementation for other applications, H.264 HW design and implementation have faced several challenges that can be sum up as main design trends.

Indeed, because of their highly efficient coding capabilities, H.264 encoders are expected to be used in challenging applications, e.g. real-time and/or high-definition video ones. For these applications, many work (e.g. [15] and references therein) aim to implement high speed HW H.264 video encoders. Due to the long coding path, H.264 encoders are mostly designed as pipeline architectures, implementing slight modifications in the entire pipeline or in some particular modules to overcome data dependency.

Data dependency appears among MBs when the current MB encoding requires the information from encoded neighboring MBs. To solve data dependency among MBs, the parallel pipelines architecture [15] or the modified motion vector prediction in the inter-prediction block [8] might be applied. Actually, the parallel pipelines architecture enables MBs to be processed in order so that all required information form neighboring MBs is available when the current MB is encoded. Thus, this method can double the encoding speed. The modified motion vector uses the motion vectors from encoded neighboring MBs, e.g. the top-left, top, and top-right instead of the left, top and top-right to predict the current motion vector. Note that data dependency also appears among tasks when the mode selection in prediction tasks needs the result of later tasks. For example, the rate control scheme requires the amount of entropy encoded data to choose the appropriate mode. Data dependency among tasks can be solved by the use of new rate control schemes [16, 17]. Other work placed the reconstruction task across the last two stages [9, 18]. The new rate control scheme calculates the cost function from the information in early tasks rather than from the entropy encoded data size.

Data dependency also requires a very high memory access rate during the coding process. Usually, an off-chip memory is used for reference frames to reduce the total area and energy cost. Then, some on-chip buffers are implemented to reduce the external bandwidth requirement, thus reducing their timing costs. For example, a local search window buffer embedded can reduce the external bandwidth from 5570 GBytes/s in the software implementation to 700 MBytes/s in the HW one [8].

Many high speed HW H.264 encoders have been proposed, as can be seen in the literature. Some of them are even able to process HDTV1080 30 fps video for real-time application [6, 15]. Meanwhile, design focusing on low-power consumption has been raised as a great challenge. Some designers tried to modify the already available architectures to reduce the memory access [18], as a result, improving the power consumption. Others used specific low-power techniques, e.g. power gating and clock gating [9, 15, 16]. Moreover, the encoder might be reconfigurable to change its profile/features so as to adapt with the power consumption requirements [9, 17].

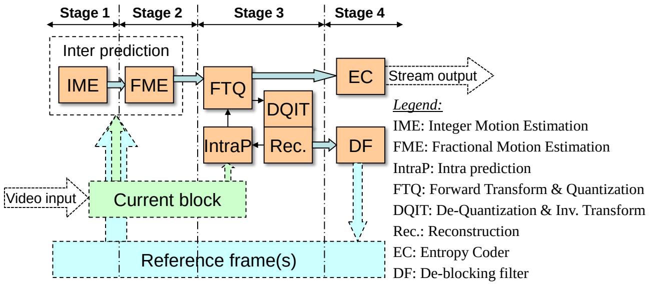

Figure 2. *Classical* 4-stage pipeline architecture of H.264 hardware encoder.

Thus, the main challenges driven by H.264 HW implementation are area cost, coding speed for real-time, high definition resolution and, of course, power consumption. With the development of semiconductor technology, the area cost drew small attention these days while researchers still focus on coding speed improvement, especially for complex encoders as specified in high profile. As video applications for mobile devices are popular nowadays, power consumption of video encoders is becoming a major concern. Next section provides a state of the art overview, various architectures being classified with respect to their main goals.

3 State of the Art and Power Feature of Different H.264 Hardware Encoders

Three main groups of HW encoder implementations found in the literature are now discussed.

3.1 H.264 Encoder Architectures

The different architectures found in literature can be classified into three main groups. The first one is a *classical* 4-stage pipelining architecture implemented since 2005 [19] and still in use in some recently published designs. The second group is a mixture of various architectures very different from the *classical* one that provide improvements in terms of coding *speed* or video *scalability*. Since the most challenging and recent problem of H.264 coding HW implementation is low-power and power aware, the last group gathers architectures with *power-oriented* designs.

3.1.1 Classical Architecture:
An H.264 encoder is typically implemented as a four-stage pipeline architecture at MB level. Figure 2 shows the major modules location in a four-stage H.264 encoder.

The Motion estimation (ME) block, operating with the Motion Compensation (MC) one to perform inter-prediction, is a potent coding tool but with a huge computational complexity. It is admitted that the ME module with full search can spend more than 90 % of the overall computation [9]. Hence, in pipelining architectures, the ME task is separated into two subtasks (i.e. integer ME (IME) and fractional ME (FME)) occupying the first two stages. To achieve a balanced schedule, the intra-prediction (Intra) is placed in the third stage. The Intra mode decision requires transform - quantization (FTQ & DQIT) and reconstruction (Rec.) in the same stage with Intra. Then, the last stage contains two independent modules, namely the entropy coder (EC) and the de-blocking filter (DF). In order to reduce the size of the buffer between stages, the pipeline is usually scheduled to operate at the MB level rather than at the frame level. The four-stage pipelining architecture cuts the coding path in a balance manner which facilitates the tasks scheduling but increases the overall latency of the encoder.

Many work implemented H.264 encoders based on this *classical* architecture, e.g. [7, 8, 16, 17]. The pipeline implemented by S. Mochizuki et al. is described to be 6-stage one [16]. However, the main encoding tasks are performed in the four middle stages in a way close to the *classical* pipeline. The ME operates in two early stages; the Intra and the transform-quantization occupy the next one; the entropy coder (VLC: variable-length coder) and the DF are placed in the remaining stage. The first and last stages are for DMA reading and writing, respectively. Moreover, the intra-prediction is modified to enable high picture quality. As specified in the H.264 standard, the "best" mode can only be selected after all predictive blocks in an MB are processed through Intra, FTQ, DQIT then Rec. In this 6-stage pipeline encoder, the mode decision part is performed before the other tasks of intra-prediction, into the previous stage. The best mode is decided from the original image but not from the locally decoded image as in the classical intra-prediction engines. With fewer logic gates and less processing time, this solution avoids limiting number of mode candidates while keeping high picture quality. With a faster Intra stage, the design in [16] is slightly less balanced than the one in [8]. Most of its improvement is provided by its special techniques, but not by the architecture.

A version of pipelining encoder containing only 3 stages is sometimes used. In this architecture, FME and Intra stages are grouped into one stage. The first advantage of this solution is that FME and Intra can share the current block and pipeline buffers [18]. Secondly, this architecture minimizes the latency on the entire pipeline [6]. Lastly, reducing the number of stages also decreases the power consumption for the data pipelining [9]. However, this pipeline obviously leads to an unbalanced schedule. When Intra and FME operate in parallel, too many tasks are put into the second stage. In order to avoid this throughput bottleneck, [9] has retimed the intra-prediction and reconstruction to distribute them into the last two stages. The luminance data is first processed in the second stage, and then the chrominance data is treated in the third one. The FME engine is also shared for the first two stages [9].

To summarize, the *classical* architecture is naturally designed from the coding path of the H.264 standard. Modifications can improve some particular features of a given design. Different architectures that remarkably improve the coding *speed* or *scalability* are now presented.

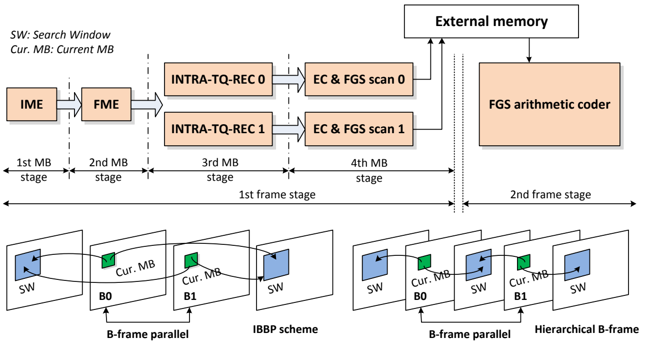

Figure 3. Two-stage frame pipeline H.264 encoder architecture.

3.1.2 Scalability and Speed-Oriented Architecture:

To achieve higher *speed* or video *scalability*, modified architectures have been proposed.

An H.264 encoder for high profile, which "firstly" supported Scalable Video Coding (SVC), was proposed in [7]. Figure 3 illustrates its 2-stage frame pipelining architecture and its B-frame parallel scheme.

The first stage contains a four-stage pipelining encoder as discussed in Section 3.1.1. The Fine-Grain-Scalability (FGS) arithmetic coder was integrated in the second pipelining stage at frame level to enable the quality scalable feature. The encoder proposed also supports spatial scalability via inter-layer prediction and temporal scalability by using Hierarchical B-frame (HB). Many schemes were adopted to reduce external memory bandwidth and internal memory access. Two consecutive B-frames can be independently encoded. The encoder processes both frames in parallel to use the common reference data. This method reduces by 50 % the external memory bandwidth of the loading searching window for GOP IBBP, and by 25 % the external memory bandwidth for the HB. In the next stages, the Intra, Rec. and EC blocks are duplicated to process two MBs concurrently. Then, the data reuse scheme in the ME engine enables both inter-layer prediction and HB while saving 70 % of the external memory bandwidth and 50 % of the internal memory access. Lastly, the FGS engine also integrates other techniques to reduce the memory bandwidth. From all these modifications and improvements, the high profile-SVC encoder, even with computation four times more complex than a baseline profile encoder, achieves comparable power consumption, i.e. only 306 mW in high profile and 411 mW with SVC for HDTV1080p video [7]. This area cost of this design can be estimated quite large when compared with classical schemes.

A high speed codec in high profile is proposed in [15]. It makes use of a two-stage pipeline encoder at frame level. The second stage operating in the stream-rate domain contains only the VLC. All other tasks are performed in the first stage in the pixel-rate domain, which contains two parallel MB-pipeline processing modules named Codec Element (CE). This method increases the processing performance but it also increases the area cost and therefore the power consumption. To support video-size scalability, on-chip connections

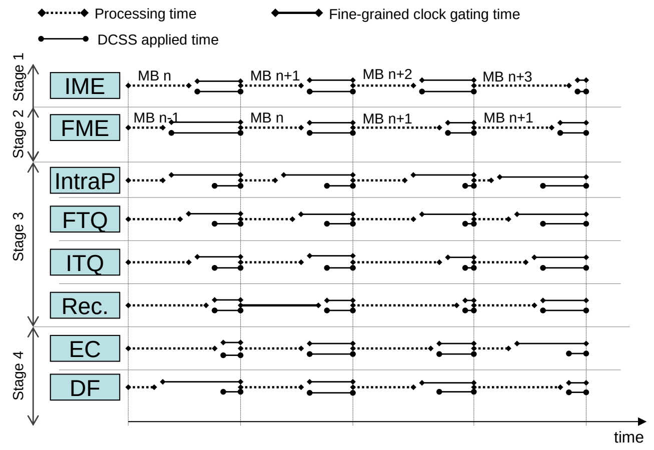

Figure 4. DCSS [16] and fine-grained clock gating [9] exploit schedule of H.264 encoder.

among sub-modules are done via a shift-register-based bus network. This bus structure enables scalability of the encoder in case more CEs are required for video-size scalability. The power consumption caused by the duplicated CE and the high computational complexity in high profile are decreased by the implementation of a Dynamic Clock-Supply-Stopping (DCSS) scheme. This low-power method will be discussed hereafter. The architecture with two parallel MB-pipelines doubles the coding throughput but it also requires extra silicon area.

As can be seen from this short review, modified architectures can provide interesting improvements with respect to the scalability and the speed features, at the price of higher power consumption or larger Silicon area, leading the designer to a tradeoff between various conflicting objectives. Power-oriented architectures that embed additional techniques to reduce the power consumption or to enable power-aware functioning are now discussed.

3.1.3 Power-Oriented Architecture: Power-oriented H.264 encoders implement the *classical* architecture with three- or four-stage pipeline together with additional low-power techniques. Among these techniques, DCSS [16] and fine-grained clock-gating [9] exploit the inactive state of sub-modules to reduce their idle power consumptions. Figure 4 shows the schedule of modules in an H.264 encoder and the time slots when power can be saved.

Actually, an unbalanced schedule will provide more opportunities to integrate low power techniques due to some inactivity phases. While DCSS cuts off the supply clock signal for the stages when all modules are not operating, clock-gating pauses the clock signal entering unused modules. DCSS was estimated to reduce up to 16 % of the power consumption [16] and fine-grain clock gating in [9] can save around 20 % in the power consumption. Thus, the latter seems to provide more power reduction but its control is more costly.

The H.264 encoder proposed in [18] does not implement the above specific low-power techniques. However, many efforts have been employed in order to reduce memory access. Firstly, Intra and FME are both placed in the second stage to use common current block and pipeline buffers. Secondly, it implements eight-pixel parallelism intra-predictor to reduce the area cost and a particular motion estimation block that can deal with high throughput. Moreover, the high throughput IME with Parallel Multi-Resolution ME (PMRME) algorithm also leads to 46 % of memory access reduction. Actually, PMRME only samples the necessary pixels to be stored in the local memory. This video encoder achieves promising power figures, that is 6.74 mW for CIF video and 176.1 mW for 1080p video in baseline profile.

A low-power ME module implementing low-power techniques is proposed in [9]. Among them are data reuse techniques to save memory access power. In the IME module, both intra-candidate data reuse and inter-candidate data reuse are applied. Intra-candidate calculates the matching cost of larger blocks by summing up the corresponding cost of smaller block (4 × 4). Inter-candidate shares overlapped reference pixels for two neighboring searching candidates. Differences among neighboring Motion Vectors (MVs) are also used to reduce the computation. In the FME block, online interpolation architecture to reuse interpolated data and mode pre-decision to reduce the number of mode candidates are adopted to save power consumption. The one-pass algorithm (and its corresponding architecture) not only alleviates the memory access but also increases the throughput of the FME sub-module. The IME data access proposed a solution which consumes 78 % less than a standard IME engine. The FME engine halves the memory access thus saving a large amount of data access power.

These designs prove that reducing memory access is an efficient high-throughput low-power scheme. However, they require the design of many specific sub-modules, which can lead to a complex and difficult design task.

Other designs propose not only the implementation of low-power techniques for the encoder but also quality scalability to improve the power consumption. Among them, the H.264 encoder proposed in [9] is dedicated to applications for mobile devices. Besides several low-power techniques as presented above, a pre-skip algorithm with a reconfigurable parameterized coding system together with a three-layer system architecture with flexible schedule enable power scalability. The pre-skip algorithm is indeed the very first step of the motion estimation module. For each MB, it compares the Sum of Absolute Differences (SAD) function of the candidate (0,0) to a threshold S in order to skip all the ME process, when possible. The parameterized coding system provides 128 different power modes based on the parameters of the IME, FME, Intra, and DF blocks. Figure 5 illustrates these parameters in the encoding system. The three-layer architecture is a hierarchical controlling system containing a system controller, a power controller and a processing engine (PE) controller. This architecture enables the clock gating technique at fine-grain level to be implemented so that the clock entering one PE can be stopped while the

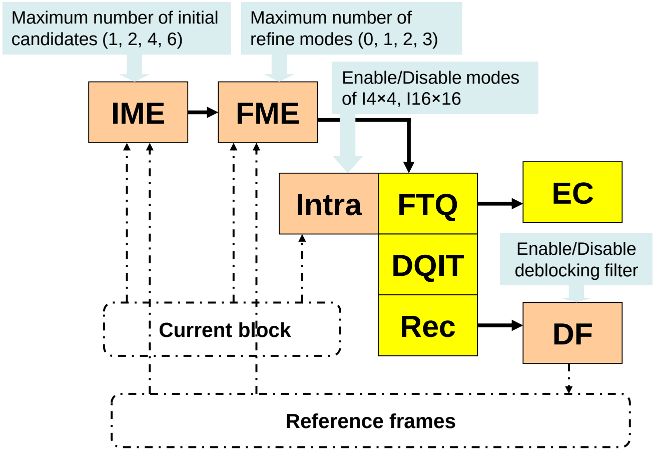
Maximum number of initial candidates (1, 2, 4, 6)
Maximum number of refine modes (0, 1, 2, 3)
Enable/Disable modes of I4×4, I16×16
IME
FME
Intra
FTQ
DQIT
Rec
EC
Enable/Disable deblocking filter
DF
Current block
Reference frames

Figure 5. Parameterized video encoder.

clock entering another PE in the same pipeline stage can be kept.

The work in [17] focuses on not only portable video applications but also a wider range of resolutions, up to HD720@30 fps. For each resolution, four different quality levels with their correspondingly power level are provided. The quality-scalability feature is implemented with parameterized modules, e.g. inter- and intra- prediction ones. Different operating clock frequencies are used in the different quality levels. Besides, some design techniques to reduce complexity of the main modules and therefore decrease their power consumption are also applied.

3.2 Discussion

Table I summarizes the state-of-the-art solutions that have been discussed in this section. Various features are presented but only the power consumption one will be discussed.

Firstly, the profile and resolution obviously influence the operating frequency and thus the power consumption. Indeed, the encoders that support multiple profiles [18] or multiple resolutions [8, 9, 16–18] operate at different frequencies and show different power consumptions. Therefore, when the power results are compared, the resolution and profile that the encoders support have to be taken into account.

Secondly, both the specific low-power techniques [9, 16] and the strategies implemented to reduce the memory access [18] present appealing power consumption figures, e.g. 9.8 – 40.3 mW for CIF video [9], 64 mW for HD720p [16] and 242 mW for 1080p high profile [18]. In the baseline profile, with 6.74 mW of power consumption for CIF video [18], the technique applied for the memory access reduction seems to perform better than the one in [9] (9.8 mW).

Lastly, recent encoders with power-aware ability [9, 17] take even less Silicon area and seem more suitable for mobile applications. With the widely admitted threshold of 100 mW of consumption for portable media applications [9], H.264 encoders for mobile devices seem to support only the baseline profile while their maximum resolution is 720HD [9, 16–18].

4 VENGME H.264 Encoder

The "Video Encoder for the Next Generation Multimedia Equipment (VENGME)" project aims at designing and implementing an H.264/AVC encoder targeting mobile platforms. The current design is optimized for CIF video; however, the architecture can be extended for larger resolutions by enlarging the reference memory and the search window.

4.1 Architecture

One of the factors which affect both computational path and the power consumption is the workload of the system and the data dependencies among the pipeline stages. In H.264/AVC encoder, the most time consuming part is inter prediction including IME, FME, and MC. The second time consuming module in the encoder is the entropy encoder (EC). Therefore, the architecture should be carefully selected to improve the coding throughput and the overall performance. In the encoding loop, intra prediction uses the reference pixels from adjacent neighbor macroblock, therefore, intra prediction have the highest in-frame data dependencies. For each intra macroblock, the current predicted macroblock must be reconstructed before predicting the next macroblock. In addition, in 4×4 prediction modes of intra prediction, each 4×4 block must be reconstructed before predicting the next 4×4 block. Because of this, intra prediction has strong relation with FTQ and the reconstructed loop. In contrast, inter prediction needs only reference pixels from the previous encoded frames. These reference data can be preloaded into search windows SRAM. Inter prediction does not need the FTQ and reconstruction loop for its prediction for next macroblock prediction. Based on this data dependency, the modules which wait for data available can be turned off to save the power consumption. For example, the FTQ/ITQ, intra prediction, reconstructed loop and the entropy encoder can be turned off when waiting for inter prediction to finished.

The architecture of VENGME H.264 encoder uses the classical 4-stage pipeline scheme with some modifications, as illustrated in Figure 6. The first stage is used to load the data needed for the prediction. It is thus similar to the architecture in [16]. The second stage includes intra- and inter-predictions. IME and FME are merged into the same stage because FME and MC can reuse the information from IME and the data from the search window SRAM. Therefore, this is different from the classical architecture, see Figure 2.

One search window SRAM and an extra external memory access bandwidth can be saved, while the performance for targeted applications remains unchanged. Inter-prediction and intra-prediction in the same stage can be executed in parallel or separately, thanks to the system controller decision. In the separate mode of execution, to save the power consumption, one of the

Table I
State of the Art: Comparison of Different H.264/AVC Encoder Architectures

Design features	**Z. Liu** [6]	**Y.-H. Chen** [7]	**K. Iwata** [15]	**T.-C. Chen** [8]	**Y.-H. Chen** [9]	**S. Mochizuki** [16]	**Y.-K. Lin** [18]	**H.-C. Chang** [17]	**H. Kim** [20]
Target	Real-time	Scalable Extension SVC; High profile	Low power; Video size scalable	Hardware design for H.264 codec	Low-power; Power aware; Portable devices	Low-power; Real-time; High picture quality	High profile; Low area cost; High throughput	Dynamic Quality-Scalable; Power-aware video applications	Low power; Power aware
Profile	Baseline, level 4	High profile; SVC	High, level 4.1	Baseline, level up to 3.1	Baseline	Baseline, level 3.2	Baseline/High, level 4	Baseline	N/A
Resolution	1080p30	HDTV 1080p	1080p30	720p SD/HD	QCIF, 720SDTV	720p SD/HD	CIF to 1080p	CIF to HD720	CIF, HD1280×720
Techno (nm)	UMC 180, 1P6M CMOS	UMC 90 1P9M	CMOS 65	UMC 180, 1P6M CMOS	TSMC 180, 1P6M CMOS	Renesas 90, 1POLY-7Cu-ALP	UMC 130	CMOS 130	N/A
Frequency (MHz)	200	120(high profile); 166 (SVC)	162	81 (SD); 180 (HD)	N/A	54 (SD); 144 (HD)	7.2 (CIF); 145 (1080p)	10-12-18-28 (CIF); 72-108 (HD720)	N/A
Gate count (KGates)	1140	2079	3745	922.8	452.8	1300	593	470	N/A
Memory (KBytes)	108.3	81.7	230	34.72	16.95	56	22	13.3	N/A
Power consumption (mW)	1410	306 (high profile); 411 (SVC)	256	581 (SD); 785 (HD)	40.3 (CIF, 2 references); 9.8-15.9 (CIF 1 reference); 64.2 (720SDTV)	64 (720p HD)	6.74 (CIF baseline); 242 (1080p high profile)	7-25 (CIF); 122-183 (HD720)	238.38 to 359.89 depends on PW level

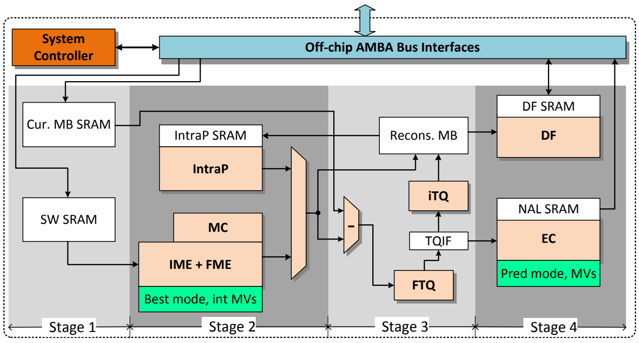

Figure 6. VENGME H.264/AVC encoder architecture.

intra- and inter-prediction can be switched off while the other is in active state. In the mixed mode of execution, the intra prediction and inter prediction can be done in parallel, the intra prediction will finish first, and its results are stored in TQIF memory. After that, the intra module can be switched off to save power. Inter prediction and motion compensation continue to find the best predicted pixels. After having inter-prediction results, TQIF memory can be invalidated to store new transformed results for inter module. The third stage and the final stage are the same as the classical 4-pipeline architecture.

Our low-cost FTQ/ITQ architecture in [13, 21] uses only one unified architecture of 1-D transform engine to perform all required transform process, including discrete cosine transform and Walsh Hadamard transform. The striking feature of this work is the fast and highly shared multiplier in the integrated quantization part inside forward transformation module. Our quantizer can saved up to 72 adders in comparison with other FTQ designs. The overall area is minimized by replacing the saved adders with multiplexers and just one 1-D transformation module with a little increase in design of the controller. Besides FTQ/ITQ, our improvements in CAVLC in [22] and then in the whole entropy encoder [23] can reduce the hardware area and the overall bit rate further. The reduction in hardware area is done by optimizing the table selector and its associated memory area with two main techniques: re-encoding VLC tables and calculating the codewords arithmetically. Furthermore, our CAVLC encoder uses zero-skipping technique with 8 × 8 block level to minimize the encoding time and lower the bit rate.

To implement the pipelining architecture, the encoder employs a double memory scheme. The current macroblock RAM and the transform and quantization interface (TQIF) contain a double memory that can store two macro-blocks at the same time. Predicted pixels from inter- and intra- modules are sent directly to the forward transform and quantization (FTQ) modules. The entropy encoding modules and the deblocking filter have their own SRAM to store information for the variable-length coding and the filtering process.

To reduce the memory bandwidth for the inter-prediction, “Snake-move” scan strategies are used so that to create new candidates, only 16 more pixels are read, as presented in [24]. The Snake-move scan strategy is illustrated in Figure 7. At first, the candidate 0 is fully read into the inter-prediction memory. Then, the next candidate is created by shift-down, shift-left or ship-up based on its position. For each new candidate for the current search window, only 16 pixels are needed. This strategy reduces the memory access to only 222 Mbytes/s for CIF video at 200 MHz.

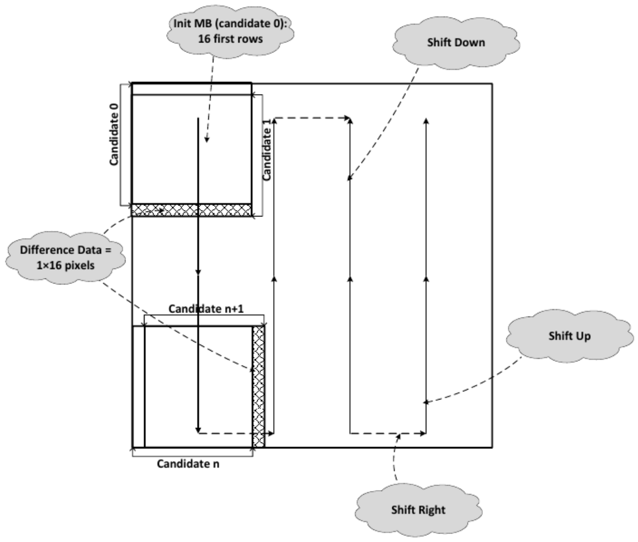

Figure 7. Snake-move scan strategy for inter prediction.

4.2 Discussion

The VENGME design is different when compared to the state of the art solutions. Besides the first stage to load data, the VENGME architecture cuts the coding path into three main stages, which are prediction, TQ-reconstruction and EC-DF. With both IME and FME in the same stage, this pipeline is even more unbalanced than the three 3-stage pipelines that can be found in the literature. However, it enables applying low-power techniques when the two last stages wait for the heaviest prediction stage. Moreover, it is not necessary to operate both intra- and inter- prediction for each MB. The system controller can decide to use the two prediction methods separately in order to reduce the task and the power consumption. This solution (currently in use) can be chosen as a low-power mode for the system.

The double memory scheme increases the area cost but it maintains the memory access. Some dedicated techniques to reduce the area cost and increase the throughput are applied to each module. For example, the snake-move scan strategy in the inter-prediction reduces the memory access, thus its timing and power consumption.

Moreover, some modules with several key contributions have been published previously [13, 21–25]. The entropy coding (EC) module contains Exp-Golomb and CAVLC coding methods. It also encapsulates all encoded video data in the Network Abstraction Layer format. The most complicated sub-module, e.g. the CAVLC encoder, implements various design techniques to reduce the processing time such as pipelining, zero-skipping, table selector integration, etc. The CAVLC encoder has been published in [22]. Its design was shown to have better throughput than the previously published ones. Actually, 5798×10^3 MBs/s can be processed while previous works can process a maximum of 738×10^3 MBs/s.

The Forward Transform and Quantization (FTQ) module implements a fast architecture of the multiplier in the most critical process, i.e., the quantizer, to increase the speed. To reduce the area cost, the design utilizes only one unified architecture of a 1-D transform engine to perform all required transform processes, i.e. a discrete cosine transform and Walsh Hadamard transform. As published in [13], this FTQ module costs only 15 Kgates, when previously published designs cost at least 23.2 Kgates. With the same 4-bit data width, the VENGME throughput is 445 Msamples/s, compared to 273 Msamples/s in previous works.

The proposed H.264 encoder has been modeled in VHDL at RTL level. The power consumption is estimated at RTL level in encoding video of QCIF resolution, with technology 32 nm, using SpyGlass^{TM} Power tool. The (estimated) total power consumption is 19.1 MW. Note that the leakage power at RTL level is not accurately estimated as it highly depends on gate choices (actually it is over-estimated). Thus, it can be assumed that this power consumption should be smaller.

In summary, a different H.264 encoder hardware pipelining architecture which enables to apply low-power techniques has been proposed. In this proposal, the throughput increase and hardware area reduction for each individual module have been considered during the design phase.

5 Conclusions and Future Works

In this paper, we have presented a survey of H.264 video encoding HW implementations. Various designs were classified into three groups of architectures. The implementations analysis focuses on power features. Classical four-stage pipelining architecture has a balanced schedule but its overall latency would be increased in comparison to the three-stage one. An unbalanced schedule may lead to bottlenecks in the encoding path; however it enables applying low-power techniques as some modules have to wait to the others' operation. Modified architectures provide significant speed or scalability improvements at the price of higher area cost and power consumption. Power-oriented architecture uses classical pipelining with additional low-power techniques or memory access reduction strategies. Parameterized designs enable power scalability and thus power-aware ability for video encoders. Power results are compared while regarding resolution and profile of the designs. Both specific low-power techniques and memory access reduction present power efficiency. Currently, the encoders for mobile applications support only baseline profile and the maximum resolution is 720HD.

We have also proposed VENGME design, a particular architecture of H.264 video encoder targeting CIF video for mobile applications. The design can be extended to higher resolutions. Our four-stage pipelining architecture has unbalanced schedule and enables applying low-power techniques. Efforts to achieve high throughput, small silicon area and low memory bandwidth were implemented in each module. The implementations of the particular modules, CAVLC and FTQ/ITQ, have been proved better than previous work in terms of throughput and area cost. Our next target is to apply low-power techniques on VENGME architecture to develop power-awareness functionality.

Acknowledgement

This work is partly supported by Vietnam National University, Hanoi (VNU) through research project No. QGDA.10.02 (VENGME).

References

[1] *ITU-T Recommendation and International Standard of Joint Video Specification. ITU-T Rec. H.264/ISO/IEC14496-10 AVC*, ITU-T Std., March 2005.

[2] *Information Technology – Coding of Audio-Visual Objects – Part 2: Visual*, ITU-T Standard Std., 1999.

[3] *Video Coding for Low Bit Rate Communication ITU-T Rec. H.263*, ITU-T Std., February 1998.

[4] *Information Technology – Generic Coding of Moving Pictures and Associated Audio Information: Video*, ITU-T Std., 1996.

[5] A. Joch, F. Kossentini, H. Schwarz, T. Wiegand, and G. Sullivan, "Performance comparison of video coding standards using lagragian coder control," in *IEEE International Conference on Image Processing (ICIP)*, 2002, pp. 501–504.

[6] Z. Liu, Y. Song, M. Shao, and S. Li, "A 1.41w H.264/AVC real-time encoder SoC for HDTV1080p," in *Proceedings of the 2007 IEEE Symposium on VLSI Circuits*, June 2007, pp. 12–13.

[7] Y.-H. Chen, T.-D. Chuang, Y.-J. Chen, C.-T. Li, C.-J. Hsu, S.-Y. Chien, and L.-G. Chen, "An H.264/AVC scalable extension and high profile HDTV 1080p encoder chip," in *Proceedings of the 2008 IEEE Symposium on VLSI Circuits*, June 2008, pp. 104–105.

[8] T.-C. Chen, S.-Y. Chien, Y.-W. Huang, C.-H. Tsai, C.-Y. Chen, T.-W. Chen, and L.-G. Chen, "Analysis and architecture design of an HDTV720p 30 frames/s H.264/AVC encoder," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 6, pp. 673–688, 2006.

[9] Y.-H. Chen, T.-C. C. Chen, C.-Y. Tsai, S.-F. F. Tsai, and L.-G. G. Chen, "Algorithm and architecture design of power-oriented H.264/AVC baseline profile encoder for portable devices," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 8, pp. 1118–1128, August 2009.

[10] S.-H. Wang, H. Y. Peng, W.-H., and *et al.*, "A platform-based MPEG-4 advanced video coding (AVC) decoder with block level pipelining," in *Proceedings of the International Conference on Information, Communications and Signal Processing*, vol. 1, 2003, pp. 51–55.

[11] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[12] I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*. New York, USA: John Willey & Sons, 2003.

[13] X.-T. Tran and V.-H. Tran, "An efficient architecture of forward transforms and quantization for H.264/AVC codecs," *REV Journal on Electronics and Communications (JEC)*, vol. 1, no. 2, pp. 122–129, 2011.

[14] M. Wen, J. Ren, N. Wu, H. Su, Q. Xun, and C. Zhang, "Data parallelism exploiting for H.264 encoder," in *Proceedings of the International Conference on Multimedia and Signal (CMSP)*, May 2011, pp. 188–192.

[15] K. Iwata, S. Mochizuki, M. Kimura, T. Shibayama, F. Izuhara, H. Ueda, K. Hosogi, H. Nakata, M. Ehama, T. Kengaku, T. Nakazawa, and H. Watanabe, "A 256mW 40Mbps Full-HD H.264 High-Profile codec featuring a dual-macroblock pipeline architecture in 65nm CMOS," *IEEE Journal of Solid State Circuits*, vol. 44, no. 4, pp. 1184–1191, 2009.

[16] S. Mochizuki, T. Shibayama, M. Hase, F. Izuhara, K. Akie, M. Nobori, R. Imaoka, and H. Ueda, "A low power and high picture quality H.264/MPEG-4 video codec IP for HD mobile applications," in *Proceedings of the IEEE Asian Solid-State Circuits Conference (A-SSCC)*, November 2007, pp. 176–179.

[17] H. Chang, J. Chen, B. Wu, C. Su, J. Wang, and J. Guo, "A dynamic quality-adjustable H.264 video encoder for power-aware video applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 12, pp. 1739–1754, December 2009.

[18] Y.-K. Lin, D.-W. Li, C.-C. Lin, T.-Y. Kuo, S.-J. Wu, W.-C. Tai, W.-C. Chang, and T.-S. Chang, "A 242mW 10mm2 1080p H.264/AVC High-Profile encoder chip," in *Proceedings of the IEEE International Solid-State Circuits Conference (ISSCC)*, February 2008, pp. 314–315.

[19] Y.-W. Huang, T.-C. Chen, C.-H. Tsai, C.-Y. Chen, T.-W. Chen, C.-S. Chen, C.-F. Shen, S.-Y. Ma, T.-C. Wang, B.-Y. Hsieh, H.-C. Fang, and L.-G. Chen, "A 1.3TOPS H.264/AVC single-chip encoder for HDTV applications," in *Proceedings of the IEEE International Solid-State Circuits Conference (ISSCC)*, vol. 1, San Francisco, CA, February 2005, pp. 128–588.

[20] H. Kim, C. E. Rhee, J.-S. Kim, S. Kim, and H.-J. Lee, "Power-aware design with various low-power algorithms for an H.264/AVC encoder," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, 2011, pp. 571–574.

[21] X.-T. Tran and V.-H. Tran, "Cost-efficient 130nm tsmc forward transform and quantization for h.264/avc encoders," in *Proceedings of the IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS)*, Cottbus, Germany, April 2011, pp. 47–52.

[22] N.-M. Nguyen, X.-T. Tran, P. Vivet, and S. Lesecq, "An efficient context adaptive variable length coding architecture for H.264/AVC video encoders," in *Proceedings of the International Conference on Advanced Technologies for Communications (ATC)*, 2012, pp. 158–164.

[23] N.-M. Nguyen, E. Beigne, S. Lesecq, P. Vivet, D.-H. Bui, and X.-T. Tran, "Hardware implementation for entropy coding and byte stream packing engine in H.264/AVC," in *Proceedings of the International Conference on Advanced Technologies for Communications (ATC)*, Ho Chi Minh City, October 2013, pp. 360–365.

[24] N.-K. Dang, X.-T. Tran, and A. Merigot, "An efficient hardware architecture for inter-prediction in H.264/AVC encoders," in *Proceedings of the 17th IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS)*, Warsaw, Poland, April 2014.

[25] D.-H. Bui, V.-H. Tran, V.-M. Nguyen, D.-H. Ngo, and X.-T. Tran, "A hardware architecture for intra prediction in H.264/AVC encoder," in *Proceedings of the IEICE International Conference on Integrated Circuits and Devices in Vietnam (ICDV)*, Danang, August 2012, pp. 95–100.

Ngoc-Mai Nguyen received the Bachelor degree in technologies of Electronics and Telecommunications from the VNU University of Engineering and Technology (VNU-UET), Vietnam National University, Hanoi, Vietnam (VNU) in 2009, and the M.Sc. degree in Information, Systems and Technology from the University Paris XI, France, in 2011. She started her work as a research engineer at the Key Laboratory for Smart Integrated Systems (SIS Laboratory) – VLSI System Design laboratory, VNU University of Engineering and Technology, Vietnam National University, Hanoi, Vietnam (VNU) from 2009.

She is currently a PhD student at the CEA-LETI, MINATEC, Grenoble, France (with Grenoble INP).

Edith Beigné received her Master degree in Microelectronics from Grenoble Polytechnical Institute in 1998. She joined CEA-LETI in 1998 focusing on asynchronous systems and circuits specifically for ultra low power mixed-signal systems. Since 2005, she is head of the low power design team within the digital laboratory developing fine-grain power control and local voltage and frequency scaling innovative features. She was leading complex innovative SoC design in 65nm, 32nm bulk and now in 28nm and 14nm FDSOI technologies for adaptive voltage and frequency scaling architecture based on GALS structures. She also works since 2009 on energy harvesting systems developing asynchronous energy-driven and event-driven platforms based on her knowledge on asynchronous event-based systems.

Duy-Hieu Bui was born in Thai Binh, Vietnam in 1988. He received a B.Sc. degree in Electronics – Telecommunication technology from VNU University of Engineering and Technology in 2010 and a M.Sc. degree in Network and Telecommunications from University of Paris-Sud XI in 2012. From 2010 to present, he has been working as a researcher at the UET-Key Laboratory for Smart Integrated Systems (SIS Laboratory) – VLSI System Design laboratory, VNU University of Engineering and Technology.

His research interests include System-on-Chip/Network-on-Chip design and verification, embedded systems, VLSI systems/circuits design for multimedia application, and fault-tolerant. He is a member of IEEE, IEICE.

Nam-Khanh Dang was born in Hanoi, Vietnam in 1989. He received a B.Sc. degree in Electronics – Telecommunication technology from VNU University of Engineering and Technology in 2010 and a M.Sc. degree in Network and Telecommunications from University of Paris-Sud XI in 2012. From 2010 to present, he has been working as a researcher at the UET-Key Laboratory for Smart Integrated Systems (SIS Laboratory) – VLSI System Design laboratory, VNU University of Engineering and Technology.

His research interests include the design and verification of System-on-Chips/Network-on-Chips, VLSI systems/circuits design for multimedia application, and fault-tolerant.

Suzanne Lesecq passed the “Agrégation” in Electrical Engineering in 1992. She received the Ph.D. degree in Process Control from the Grenoble Institute of Technology (Grenoble INP), France, in 1997. She joined the University of Grenoble, France in 1998 where she has been appointed as Associate-Professor from 1998 to 2006 and full-time Professor from 2006 to 2009. She joined CEA-LETI in mid 2009. She has published more than 90 papers in world leading Conferences, International Journals, book chapters.

Her topics of interest are data fusion and control theory, together with their safe implementation on computational devices.

Pascal Vivet graduated from Telecom Bretagne, Brest and received his Master of Microelectronics from University Joseph Fourier (UJF), Grenoble in 1994. He accomplished his PhD in 2001 within France Telecom lab, Grenoble, designing an asynchronous Quasi-Delay-Insensitive microprocessor in the research group of Pr. Marc Renaudin. After 4 years within STMicroelectronics, Pascal Vivet has joined CEA-Leti in 2003 in the advanced design department of the Center for Innovation in Micro and Nanotechnology (MINATEC), Grenoble, France.

His topics of interests are covering wide aspects from system level design and modeling, to asynchronous design, Network-on-Chip architecture, low power design, many core architectures, 3D design, including interactions with related CAD aspects. On 3D architecture, he strongly participates to the 3D design and roadmap within LISAN lab, with contributions on 3D asynchronous NoC, 3D Thermal modeling, 3D Design-for-Test, 3D physical design, and coordination of corresponding 3D CAD tools and CAD partners. Dr Pascal Vivet participates to various TPC such as ASYNC, NOCS, DATE, 3DIC conferences. He was general chair of ASYNC’10, and program chair of ASYNC’12. He was program chair of 3D workshop at DATE’14, and program chair of D43D’13 and D43D’14 workshops. He is the author or co-author of a couple of patents and of more than 50 papers.

Xuan-Tu Tran received a B.Sc. degree in 1999 from Hanoi University of Science and a M.Sc. degree in 2003 from Vietnam National University, Hanoi, all in Electronics Engineering and Communications; and a Ph.D. degree in 2008 from Grenoble INP (in collaboration with the CEA-LETI), France, in Micro Nano Electronics. Xuan-Tu Tran is currently an associate professor at the VNU University of Engineering and Technology (VNU-UET), a member university of Vietnam National University, Hanoi (VNU). He is currently Deputy Director of UET-Key Laboratory for Smart Integrated Systems (SIS Laboratory) and Head of VLSI Systems Design laboratory. He is in charge for CoMoSy, VENGME, ReSoNoC projects for embedded systems and multimedia applications. His research interests include design and test of systems-on-chips, networks-on-chips, design-for-test, VLSI design, low power techniques, and hardware architectures for multimedia applications.

He is a Senior Member of the IEEE, IEICE, and the Executive Board of REV. He serves as Vice-Chairman of IEICE Vienam Section, Chairman of IEEE SSCS Vietnam Chapter. He also served as chair/co-chair and organizing/technical committee member for numerous international conferences, Associate Editor-in-Chief of REV Journal on Electronics and Communications (JEC), Editor of VNU Journal of Computer Science and Communication Engineering (JCSCE). He is the author or co-author of more than 50 papers and a monograph.