POWER-SCALABLE ALGORITHM AND RECONFIGURABLE MACRO-BLOCK PIPELINING ARCHITECTURE OF H.264 ENCODER FOR MOBILE APPLICATION

Yu-Han Chen, Tung-Chien Chen, and Liang-Gee Chen

DSP/IC Design Lab, Graduate Institute of Electronics Engineering and
Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan
Email:{doliamo, djchen, lgchen}@video.ee.ntu.edu.tw

ABSTRACT

In this paper, a power-scalable H.264 encoding system is provided with the efforts on both the algorithm and the architecture levels. For a start, a Motion Estimation (ME) pre-skip algorithm is adopted as a system-level power-scalable algorithm. In order to realize a dedicated hardware, a novel reconfigurable Macro-Block (MB) pipelining architecture is proposed. It can improve not only system flexibility but also hardware efficiency. Besides, it is also beneficial for power management with module-level gated clock insertion. According to simulation results, the proposed H.264 encoder can support power-scalable functionality in the range of about 20 to 90 mW with graceful quality degradation.

1. INTRODUCTION

Enjoying multimedia services on mobile devices is a key application nowadays. Due to the limited battery capacity, low-power design becomes much more important in recent years. Power-aware computing [1] is an emerging concept which can adjust the power consumption in response to different operating conditions. For example, in a power-rich environment, a high quality service is preferred in spite of the cost of higher power consumption. On the contrary, if the battery capacity is low, users may tolerate poor quality services with lower power consumption in order to extend the battery lifetime.

Fig. 1 shows the performance of a power-aware video coding system. The quality above the dotted line is saturated, i.e., increasing a large amount of power only contributes to little quality improvement. *Point A* is a power-efficient operating point for the high quality service. *Point B* is the ultra-low-power point. Between *Point A* and *Point B*, there are two power-aware curves—the solid one and the dashed one. The solid curve is the better one because quality is degraded more gracefully as power is decreased.

Owing to outstanding coding efficiency and visual quality, H.264 [2] is a potential video coding standard for mobile applications. Powerful coding tools, such as Variable Block-Size (VBS), Multiple Reference Frames (MRF), Intra Prediction (IP), CAVLC, and in-loop De-Blocking (DB) filter, en-
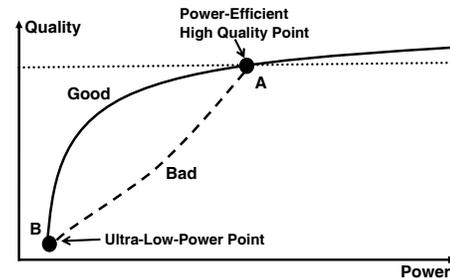

Fig. 1. Power-aware curve.

hance the coding gain of H.264 but lead to high computation and bandwidth overhead, which both induce high power consumption. In a low-power design, tools with high computation may be discarded or simplified. Coding performance is sacrificed for power reduction, and it makes the main advantage of H.264 disappear. Therefore, power-aware computing is more applicable to an H.264 design. According to the power constraint and the video contents, some suitable coding tools are chosen to maintain coding performance, and the others are turned-off to reduce power consumption.

In [3], a processor-base power-scalable video encoder is provided. Authors develop a parametric video encoding architecture, which is fully scalable in computational complexity. By use of Dynamic Voltage Scaling (DVS), power-aware computing can be achieved. However, this design is not a low-power solution because processor-based designs are not power-efficient. In order to achieve both low-power and power-aware, a dedicated hardware for a power-scalable encoder is required.

According to the power profile in Fig. 2, more than 70% power of the whole H.264 encoder is consumed by the ME engine. On the algorithm level, a ME pre-skip algorithm is adopted in this paper to provide power-scalable functionality. Power consumption of the whole encoding system can be controlled with the number of pre-skipped MBs. The idea of checking the SKIP mode in the early stage was first proposed in [4]. However, previous works all focus on reducing computation and maintaining coding performance. Thus only *Point A* in Fig. 1 is supported. On the contrary, the proposed ME
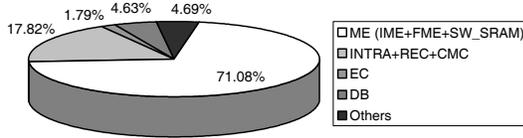
ICME 2006

Fig. 2. Power profile of H.264 encoder [5] with CIF format, 2 reference frames, and H[-32, 32) and V[-16, 16] searching range.

pre-skip algorithm can provide the whole power-aware curve.

In order to realize the pre-skip algorithm in the hardware design, a novel reconfigurable MB-pipelining architecture is proposed to improve system flexibility. Besides, module-level gated clock is easily adopted in this architecture, and each module can be shut down immediately once it enters the idle state. Therefore, power efficiency is greatly improved. Finally, the proposed H.264 encoder can support power-scalable functionality in the range of about 20 to 90 mW with graceful quality degradation.

The rest of this paper is organized as follows. The research problems of developing a power-scalable H.264 encoder are described in Section 2. Section 3 and Section 4 are the power-scalable algorithm and the reconfigurable system architecture, respectively. Simulation results are shown in Section 5, and we will draw a conclusion in Section 6.

2. PROBLEM DEFINITION

How to utilize power efficiently is an important issue in a power-limited environment. A video frame can be divided into two parts—the simple motion region and the complex motion region. The motion of the MBs in the simple motion region, like the background, is easy to be predicted. Lots of computational power can be saved there without much quality degradation. On the contrary, the MBs in the complex motion region need more searching efforts to maintain coding performance. Therefore, a content-aware ME algorithm is good to achieve a balance between quality and power.

How to develop a system-level power-aware algorithm is another problem in this paper. Previous works [6] [7] focus on power-aware module designs but not the whole encoding systems. As we know, the most power-consuming part in an H.264 encoder is ME, including Integer ME (IME) and Fractional ME (FME). Therefore, a power-scalable algorithm focusing on the ME mode decision flow is a must.

For realizing a power-scalable encoder, how to develop a flexible system architecture is the main challenge. A processor-based design can provide the most flexibility, but it is not a power-efficient system. Conversely, the conventional dedicated hardware may achieve lower power consumption but lacks flexibility. Consequently, a reconfigurable architecture which can automatically configure the hardware to fit different requirements is a good candidate. Previous works mainly focus on reconfigurable module designs [8] [9]. However, a
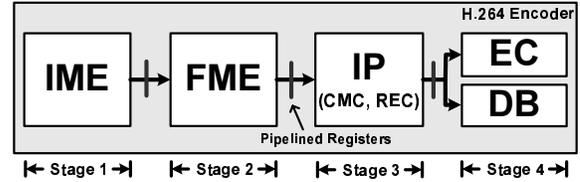

Fig. 3. Conventional H.264 4-stage MB pipelining [5].

reconfigurable system architecture is required in this paper.

Improving power efficiency is essential to a power-driven design. Module-level gated-clock insertion is beneficial for power management because Processing Engines (PE) can be shut down in the idle state by use of this technique. Previously, a four-stage MB pipelining for an H.264 encoder is proposed [5] and shown in Fig. 3. In $Stage$ 3, IP, Chroma Motion Compensation (CMC), and REConstruction (REC) engines are combined into a pipeline stage. Thus the input clock of $Stage$ 3 cannot be gated until all the PEs are idle, and much idle power will be wasted. Therefore, how to design a power efficient system architecture is important here.

3. POWER-SCALABLE ALGORITHM

3.1. Content-Aware ME Pre-Skip Algorithm

The movement of a MB in the simple motion region can be well predicted by the Motion Vectors (MVs) of the neighboring MBs. Besides, MVs in a still background are usually $(0, 0)$. In H.264, SKIP mode is defined to reduce the coding bits when the conditions above are met. If a MB is skipped, only one MV is used for the 16×16 MB, either $(0, 0)$ or the Motion Vector Predictor (MVP). Therefore, the two possible skip MVs can be pre-checked. If the skipped MBs are found in the early stage, computation of ME can be saved. There are many skipped MBs (40–70%) in the low bitrate condition, and thus computation can be greatly reduced with the pre-skip checking strategy.

Before the process of ME, 16 4×4 SAD (Sum of Absolute Difference) costs of possible skip MVs are compared to a predefined skip THreshold (TH). If all the SAD costs are less than the skip TH, this MB is pre-skipped. In the simple motion region, MBs are usually predicted well by the skip MVs and easy to be pre-skipped. Conversely, MBs in the complex motion region can hardly pass the pre-skip criterion because the SAD costs are usually too high. With the content-aware strategy, power can be efficiently utilized.

At last, the higher the skip TH is, the more MBs will be pre-skipped, and thus the power consumption is reduced. According to different power constraints, the skip TH is adjusted to achieve power-aware computing.
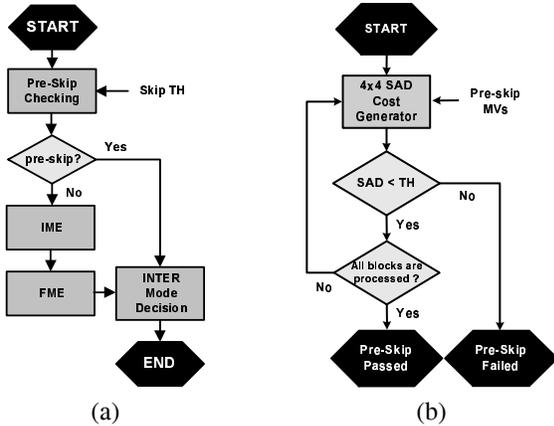
Fig. 4. Proposed ME pre-skip mode decision flow. (a) Modified ME mode decision flow; (b) 4×4 pre-skip checking procedure.

3.2. ME Pre-Skip Mode Decision Flow

The proposed ME pre-skip mode decision flow is illustrated in Fig. 4 (a). $Pre\text{-}Skip\ Checking$ is inserted at the start of the mode decision flow. If a MB is pre-skipped, ME engines are turned off, and $INTER\ Mode\ Decision$ is directly preformed. Otherwise, IME and FME are sequentially used to find the best MV. By the way, fast IME [10] and FME [11] algorithms are used to reduce power in the proposed design, but it is not the key point of this paper.

Fig. 4 (b) shows the pre-skip checking procedure. 4×4 $SAD\ Cost\ Generator$ generates the SAD costs of the pre-skipped MVs—(0, 0) and MVP. Then, the costs are compared to the skip TH sequentially. Once a SAD cost is larger than the skip TH, pre-skip checking is early terminated and failed. Otherwise, pre-skip checking is passed. Because pre-skip checking is an additional load of the whole ME mode decision flow, early termination can reduce the overhead.

4. RECONFIGURABLE MB PIPELINING ARCHITECTURE

In the conventional MB pipelining architecture, pipeline controls and pipelined registers are combined with PEs (as shown in Fig. 3). The intermediate data in a MB pipeline are restricted to be accessed by the PEs at the same MB pipeline stage. That is to say, PEs cannot access the data in different pipeline stages, and hardware flexibility is restricted. Here comes an example. Two possible pre-skip MVs need to be pre-checked in the proposed ME pre-skip mode decision flow. Among them, MVP may be a fractional MV. It means that FME engine is required to be operated in the first pipeline stage for pre-skip checking. However, it violates the pipeline order in Fig. 3. Besides, one PE cannot be used in different pipeline stages under that architecture. Therefore, the pre-skip checking algorithm is not applicable in the conventional
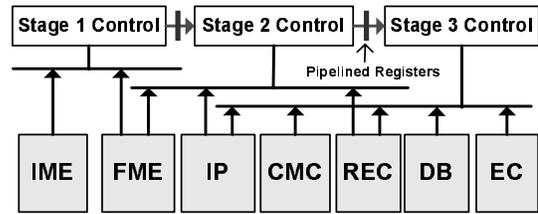

Fig. 5. Proposed reconfigurable MB pipelining.

MB pipelining architecture.

To solve this problem, the proposed system architecture in Fig. 5 is divided into two parts—pipeline stage controls and PEs. The stage control handles the MB pipeline schedule and generates control signals to assign tasks to PEs. At the end of each task, the intermediate data are stored back from PEs to pipelined registers of the stage control. Then, the stage control can assign a task and transfer the required intermediate data to other PEs. By means of this architecture, one PE is not restricted to be operated only in one pipeline stage, and the ME pre-skip mode decision flow can be supported.

At last, PEs are clearly separated from each other under the system architecture. When the task of a PE is finished, the input clock can be gated immediately with the module-level gated clock technique. As a result, power will not be wasted in the idle state. Therefore, the proposed MB pipelining can improve not only system flexibility but also power efficiency.

Different standards or specifications may be suitable to different system schedules. By means of the proposed architecture, each PE can be freely used in different pipeline stages, and a reconfigurable system schedule can be achieved. That is to say, the proposed reconfigurable MB-pipelining architecture is potential to be adopted in the hardware design of a multi-standard coder or a transcoder.

5. SIMULATION RESULTS

The proposed power-scalable algorithm is embedded into the H.264 reference software [12]. The input control parameter is the percentage of the pre-skipped MBs. The skip TH is on-line adjusted to pre-skip the required amount of MBs. At most 2 Reference Frames (RF) are supported here because MRF leads to high power overhead, and quality is usually saturated at 2 RFs. Besides, all other coding tools in Baseline Profile are supported.

Fig. 6 shows the R-D curves. The coding performance includes 2 RFs, 1 RF, and 1 RF with 25%, 50%, and 75% pre-skipped MBs. As we can see, the R-D curves of the performance with 1 RF without pre-skipped MBs and 1 RF with 25% pre-skipped MBs are nearly overlapped. Even the performance with 50% pre-skipped MB is close to them. That is to say, the content-aware ME pre-skip algorithm can utilize computational resources efficiently. As a result, quality is gracefully traded with computation as the number of pre-skip MBs is increasing.
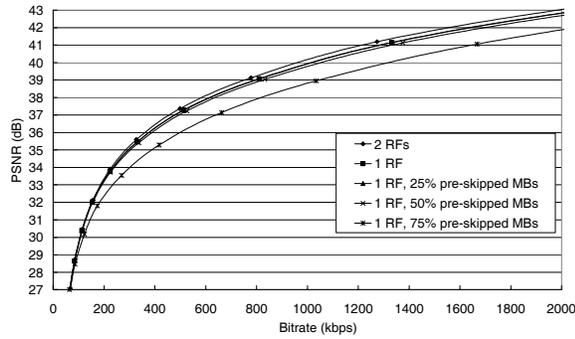
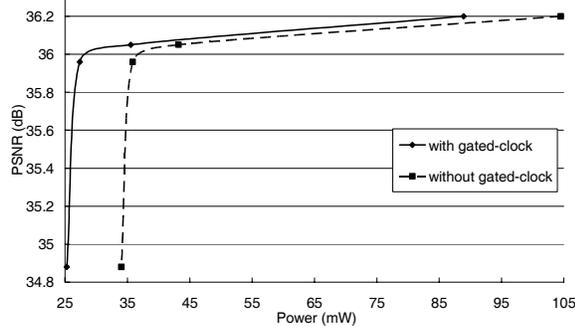**Fig. 6**. R-D Curve of $Foreman$ with CIF format.

Fig. 7. Power-aware curves of $Foreman$ with CIF format at 384 kbps.

We have realized the proposed power-scalable H.264 encoder with 452k logic gates and 16 kbytes on-chip SRAM. The required operating frequency is 27 MHz for 2 RFs and 13.5 MHz for 1 RF. With different number of pre-skipped MBs and reference frames, our design can support a power-aware curves in Fig. 7. The power data are generated from gate-level simulation. Four operating points on the curve from high quality to low quality are 2 RFs, 1RF with 25%, 50%, and 75% pre-skipped MBs, respectively. The operating point of 1 RF without pre-skipped MBs is not power-efficient and not shown. Besides, the power reduction from module-level gated-clock insertion is also shown in Fig. 7. About 8 mW idle power for 1 RF and 16 mW for 2 RFs can be saved in this design.

6. CONCLUSION

A power-scalable H.264 encoder is introduced in this paper. At first, a ME pre-skip mode decision flow is proposed as the power-scalable algorithm. By means of the content-aware strategy, power can be efficiently utilized. Then, a novel reconfigurable MB pipelining architecture is presented. With the flexible system architecture, the power-scalable functionality can be supported, and hardware efficiency is improved. According to the simulation results, the H.264 encoder can support power-aware computing in the range of about 20 to 90 mW with graceful quality degradation.

7. REFERENCES

[1] M. Bhardwaj, R. Min, and A. P. Chandrakasan, "Quantifying and enhancing power awareness of VLSI systems," *IEEE Trans. VLSI Syst.*, vol. 9, pp. 757–772, Dec. 2001.

[2] Joint Video Team of ITU-T and ISO/IEC JTC 1, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 — ISO/IEC 14496-10 AVC)," Mar. 2003.

[3] Zhihai He *et al.*, "Power-rate-distortion analysis for wireless video communication under energy constraints," *IEEE Trans. CSVT*, vol. 15, pp. 645–658, May 2005.

[4] J. Lee and B. Jeon, "Fast mode decision for h.264," *Proc. IEEE Int. Conf. Multimedia Expo*, 2004.

[5] Y.-W. Huang *et al.*, "A 1.3TOPS H.264/AVC single-chip encoder for HDTV applications," *Proc. IEEE International Solid-State Circuits Conference*, 2005.

[6] Rei-Chin Ju *et al.*, "A parameterized power-aware IP core generator for the 2-D 8x8 DCT/IDCT," *Proc. IEEE Int. Symp. Circuits Syst.*, 2004.

[7] H.-W. Cheng and L.-R. Dung, "A Vario-power ME architecture using content-based subsample algorithm," *IEEE Transactions on Consumer Electronics*, vol. 50, pp. 349–354, Feb. 2004.

[8] J. Park and K. Roy, "A low power reconfigurable DCT architecture to trade off image quality for computational complexity," *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, 2004.

[9] R. Gao, D. Xu, and J. Bentley, "Reconfigurable hardware implementation of an improved parallel architecture for MPEG-4 motion estimation in mobile applications," *IEEE Transactions on Consumer Electronics*, vol. 49, pp. 1383–1390, Nov. 2003.

[10] Y.-H. Chen, T.-C. Chen, and L.-G. Chen, "Hardware oriented content-adaptive fast algorithm for variable block-size integer motion estimation in H.264," *Proc. IEEE Int. Symp. on Intelligent Signal Processing and Communications Systems*, 2005.

[11] T.-C. Chen, Y.-W. Huang, and L.-G. Chen, "Fully utilized and reusable architecture for fractional motion estimation of H.264/AVC," *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, 2004.

[12] "H.264/AVC reference sofware JM8.2," July 2004.