PAPER *Special Section on Low-Power System LSI, IP and Related Technologies*

A Feed-Forward Dynamic Voltage Control Algorithm for Low Power MPEG4 on Multi-Regulated Voltage CPU

Hideo OHIRA[†a)], ***Nonmember***, **Kentaro KAWAKAMI**[†], **Miwako KANAMORI**[†], **Yasuhiro MORITA**[†],
Masayuki MIYAMA[†], ***and*** **Masahiko YOSHIMOTO**[†], ***Members***

SUMMARY In this paper, we describe a feed-forward dynamic voltage/clock-frequency control method enabling low power MPEG4 on multi-regulated voltage CPU with combining the characteristics of the CPU and the video encoding processing. This method theoretically achieves minimum low power consumption which is close to the hardware-level power consumption. Required processing performance for MPEG4 visual encoding totally depends on the activity of the sequence, and high motion sequence requires high performance and low motion sequence requires low performance. If required performance is predictable, lower power consumption can be achieved with controlling the adequate voltage and clock-frequency dynamically at every frame. The proposed method in this paper is predicting the required processing performance of a future frame using our unique feed-forward analysis method and controlling a voltage and frequency dynamically at every frame along with the forward analysis value. The simulation results indicate that the proposed feed-forward analysis method adequately predicts the required processing performance of every future frame, and enables to minimize power consumption on software basis MPEG4 visual encoding processing. In the case that CPU has Frequency-Voltage characteristics of 1.8 V @400 MHz to 1.0 V @189 MHz, the proposed method reduces the power consumption approximately 37% at high motion sequences or 65% at low motion sequences comparing with the conventional software video encoding method.

key words: *MPEG4 encoder, low power, feed-forward voltage control, multi-regulated voltage CPU*

1. Introduction

The 3rd generation wireless communication services have been started, and rich media services, mainly audio/visual communication or streaming services, have been expected to be a key application through mobile phone terminals. The visual communication processing requires high processing performance around several hundred MOPS, therefore dedicated hardware approach has been a major approach in terms of low power advantages on mobile terminals [1], [2]. Another recent approach is dual-processor approach [3] (CPU+DSP). The advantage of dual-processor approach achieves acceptable power consumption and flexibility. However, compared with the hardware MPEG4 LSI or dual-processor approach, software based application with single CPU has much advantages in terms of its architectural simplicity, cost effectiveness, flexibility, and extensibility for future system. Recent emerge of CPU [4] for mobile terminals has an architecture of low power and high performance using a technology of multi-regulated voltage technique. This technology achieves less power consumption than conventional CPU, however power consumption of MPEG4 encoding is still beyond acceptable power consumption. To concur the power consumption issue at software-based MPEG4 encoding, we propose in this paper dynamic voltage/clock-frequency control method targeting to achieve hardware-level power consumption.

In this paper, we describe the method to reduce power consumption for software-video encoding combining the characteristics of the CPU and the video encoding processing targeting to achieve theoretically minimum low power consumption which is closely equivalent to the hardware-level power consumption. In Sect. 2, MPEG4 visual processing performance and its characteristics are described. Video processing performance dynamically depends on the motion activity. Two processing functions, motion compensation and IDCT, are described whose processing performance are dynamically changed. In Sect. 3, recent trend of multi-regulated voltage CPU technology is briefly mentioned, and voltage/clock-frequency control method which gives the lowest power consumption in theoretical is also explained. A Dynamic voltage/clock-frequency method by a proposed forward analysis for MPEG4 processing which predicts the adequate voltage/clock frequency at every frame is introduced in Sect. 4. The simulation results are shown in Sect. 5. The simulation results indicate that the dynamic voltage/clock-frequency method by our proposed forward analysis adequately predicts the required processing performance, and enable to minimize power consumption on software basis MPEG4-visual encoding processing. In the case that CPU has characteristics of running on 1.8 V@400 MHz, 1.0 V@189 MHz the proposed method reduces the power consumption approximately 37% at high motion sequences or 65% at low motion sequences comparing with the conventional software video encoding method.

2. Processing Performance for MPEG4 Visual Encoding

Figure 1 shows the performance required for MPEG4 simple profile encoding and decoding. Approximately 200–300 MOPS are required for MPEG4 QCIF 15 fps encoding. However, this value in the figure are the average value and high motion sequence requires more performance, and low motion sequence requires less performance. Required performance depends totally on the video sequence activity. At

Manuscript received September 3, 2003.
Manuscript revised December 25, 2003.

[†] The authors are with Kanazawa University, Kanazawa-shi, 920-8667 Japan.
a) E-mail: hohira@nvidia.com

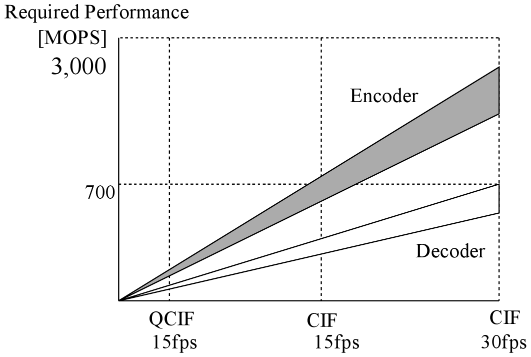


Fig. 1 MPEG-4 performance.

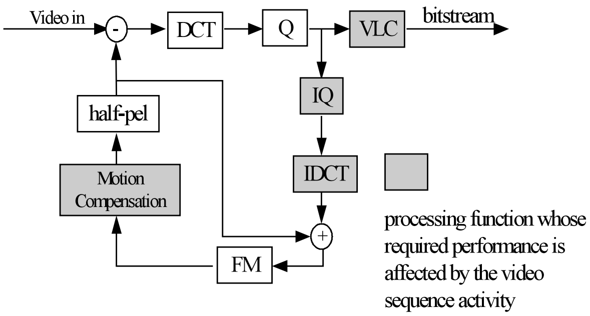

Video in
-
DCT
Q
VLC
bitstream
IQ
half-pel
IDCT
Motion Compensation
+
FM
processing function whose required performance is affected by the video sequence activity

Fig. 2 MPEG4 processing block diagram.

MPEG4 processing shown in Fig. 2, the shaded blocks are the processing function whose performance is affected by video sequence activity. Required performance of motion compensation (or MC), Inverse DCT (IDCT), Inverse Quantization (IQ), and Variable Length Coding (VLC) has been influenced according to the characteristics of the video sequence. Each processing is computationally intensive function, and approximately eighty percent of total MPEG4 performance is occupied by MC, IDCT, IQ, and VLC. Consequently, total required MPEG4 processing performance completely varies according to the sequence activity. The following sub-section describes two processing functions whose processing performance changes dynamically.

2.1 Dynamic Processing Performance Range of Motion Compensation

For motion compensation technique for software based approach, three step-search (TSS) is very well-known approach to reduce processing performance. TSS algorithm is shown at Fig. 3. At Fig. 3, in the case that search range is ±16, 49 points are selected as block matching point at 1st searching layer. 8 points are selected and each distortion value is calculated for block matching at 2nd searching layer. At 3rd layer, also 8 points are selected and calculated distortion value. After 3 layer search, the point which gives minimum distortion value at 3rd layer is chosen as best matched block. Totally, 65 distortion values (49 points at first layer and 8 points at both 2nd and 3rd layer) are calculated using block matching technique for TSS motion compensation. In addition to the TSS algorithm, a threshold technique in TSS is introduced in our simulation to reduce the processing power according to the sequence activity. A threshold technique is to skip the rest of the block-matching when the distortion value of block matching is less than threshold value. Using this technique, processing performance of TSS with threshold technique processing performance (search range: ±16) ranges from approx. 1 MOPS to 250 MOPS @QCIF 15 fps depending on the sequence activity without any critical subjective distortion error.

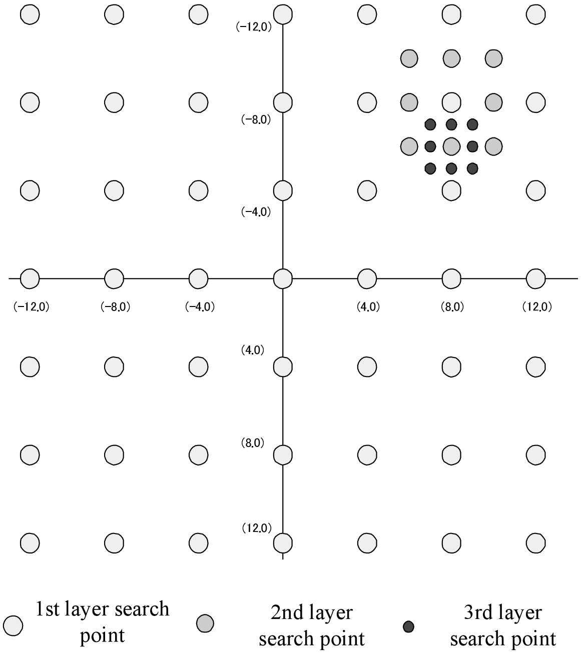


Fig. 3 TSS algorithm.

2.2 Dynamic Processing Performance Range of IDCT

IDCT processing in MPEG4 is defined as

$$f(x,y) = 1/4C(u)C(v)\sum_{u=0}^{7}\sum_{v=0}^{7}F(u,v) \times \cos((2x+1)u\pi/2N)\cos((2y+1)v\pi/2N) \qquad (1)$$

where

$F(u,v)$ = input data

$C(u), C(v)$: $1/\sqrt{2}$ when $u = 0$ or $v = 0$

$C(u), C(v)$: 1 when $u \neq 0$ and $v \neq 0$

Equation (1) is processed by 2D matrix operation as defined in Eq. (2) on the processor.

$$F(u,v) = AxA^t \qquad (2)$$

where

$$A = \begin{pmatrix} a_{00} & a_{01} & a_{02} & \cdots & a_{07} \\ a_{10} & a_{11} & a_{12} & \cdots & a_{17} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{70} & a_{71} & a_{72} & \cdots & a_{77} \end{pmatrix}$$

A: coefficient matrix

$$X = \begin{pmatrix} x_{00} & x_{01} & x_{02} & \cdots & x_{07} \\ x_{10} & x_{11} & x_{12} & \cdots & x_{17} \\ \vdots & \vdots & \vdots & & \vdots \\ x_{70} & x_{71} & x_{72} & \cdots & x_{77} \end{pmatrix}$$

X: input data matrix

An 8 by 8 2-D matrix operation requires 1024 times of multiply & accumulation (MAC) operations or 6.0M times of MAC operations per frame. However, when all the coefficients of block is zero, then IDCT processing can be skipped. Low motion sequences have more zero value and zero block, and high motion sequences have less zero blocks which eventually cause the variety of calculation power range. The range is from 0 to approx.20 MOPS at the case of QCIF 15 fps depending on the sequence activity.

3. Multi-Regulated Voltage CPU Technology

Recent break-through for low power and high performance CPU is multi-regulated voltage CPU technology. The conventional CPU runs with fixed voltage and fixed clock-frequency. On the other hand, the multi-regulated voltage/clock-frequency CPU runs with high frequency on high voltage, and runs with low frequency on low voltage. The value of voltage and clock frequency can be controlled dynamically from the software applications on CPU. The definition of energy consumption E is as follows;

$$E = \alpha \times C \times F \times V^2 \times t \tag{3}$$

α: coefficient, C: numbers of transistors,
F: clock frequency, V: voltage, t: time duration.

Here let us assume to perform "H" cycles at the duration of "T." Figure 4 shows an example of different voltage/clock frequency control method. Both method (A) and (B) have capability to achieve "H" clock cycles. The method (A) is to control clock frequency as constant F ($F = H/T$) at the duration of "T." The method (B) is to control clock frequency using two level value, i.e., F' ($F' = 2H/T$) at first half of $T/2$, and zero at the later half of $T/2$. At the case that "H" is 150 MHz cycle, F is set to 150 MHz, F' (=2F) is set to 300 MHz. $V(F)$ and $V(2F)$ is assumed to set to 0.9 V and 1.5 V respectively.

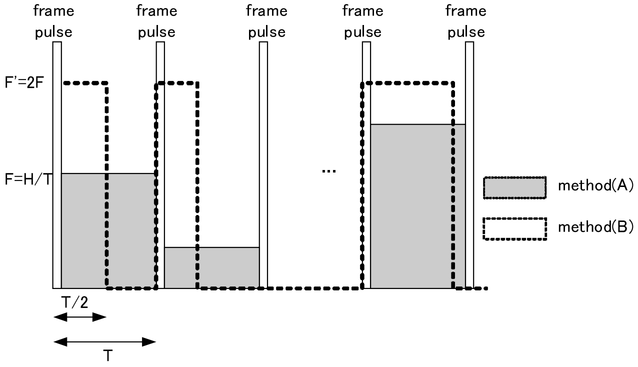


Fig. 4 Voltage/clock control method.

Substituting these value to Eq. (3),

$$\frac{E(a)}{E(b)} = \left[\frac{V(2F)}{V(F)}\right]^2 = \left[\frac{0.9}{1.5}\right]^2 = 0.36 \tag{4}$$

Equation (4) indicates that method (A) consumes only 36% of power consumption comparing with method (B) even though the performance of both method (A) and method (B) are "H" clock cycles. From this result, flat-voltage ($V = V(H/T)$) control for all the allocated duration gives lowest power consumption in theoretical [5].

4. Dynamic Voltage/Clock-Frequency Control Method by Forward Analysis

As mentioned above, the performance of MPEG4 dynamically changes according to the sequence activity. For targeting to enable low power MPEG4 on software basis, some approaches have been done by controlling voltage/clock-frequency dynamically with feed-back method at every several MacroBlocks (MBs) [6]. However, at software based approach, minimum unit of time constraint is a frame. As led from the result in Sect. 2, controlling voltage/clock-frequency dynamically by every frame theoretically minimizes power consumption. The conventional feed-back approach is not able to control voltage by frame, as prediction of required performance is necessary. Our approach is a feed-forward voltage/frequency control method dynamically by frame predicting the required performance using our unique feed-forward analysis method.

4.1 Details of Our Proposed Forward Analysis Method

A Block-diagram of forward analysis is shown in Fig. 5, and Fig. 6 shows the timing sequence of MPEG4 processing and forward analysis. The process of MPEG4 encoding using forward-analysis is as follows;

1) Prediction of the required performance for MPEG4 encoder per frame using the parameters of motion activity or other parameters.
2) Calculation of the required clock-frequency from the forward-analysis prediction.
3) Controlling the CPU voltage and clock-frequency from the software running on CPU.
4) Encoding of the new frame at the modified clock-frequency and CPU voltage.

The above 1)–3) processing corresponds to the forward analysis method, and it required only less than 1 MHz cycle, and negligible compared to the MPEG4 encoding. The duration time for voltage/clock stability after controlling voltage value ("B" in Fig. 6) is μs order which is also negligible comparing to the allocated time for frame (ex. Allocated time for a frame in case of 15 fps is 66.7 ms).

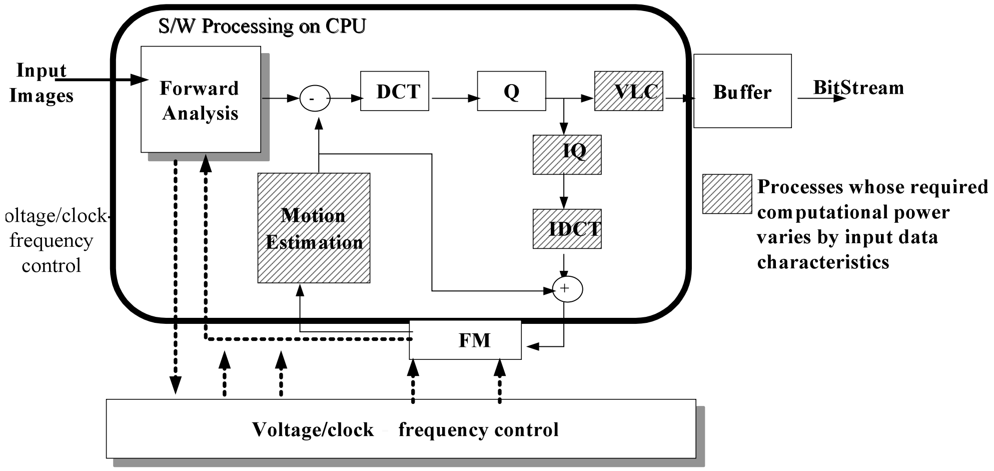
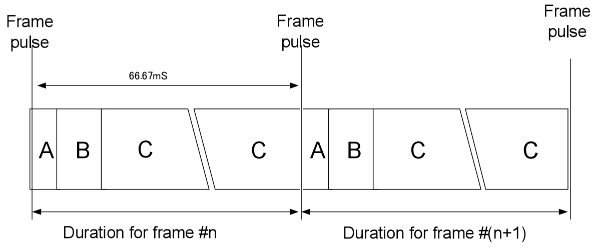

S/W Processing on CPU
Input Images
Forward Analysis
-
DCT
Q
VLC
Buffer
BitStream
IQ
IDCT
+
Motion Estimation
oltage/clock frequency control
FM
Voltage/clock frequency control
Processes whose required computational power varies by input data characteristics

Fig. 5 Block-diagram of feed-forward dynamic voltage/clock-frequency control method.



A : Forward Analysis Processing (~0.1ms)
B: Duration for Voltage/Clock stability(~60 μ s)
C: MPEG4 Processing(66.6ms)

Fig. 6 Timing sequence for forward analysis.

Table 1 Processing function.

Processing function	Affecting parameters for the processing power	Parameters
Motion compensation	Number of MB matching	N
Inverse Quantization	Numbers of coefficient	VC
Inverse DCT	numbers of valid block	VB
Variable Length Coding (VLC)	numbers of valid block	VB

4.2 Prediction of Required Performance for Forward-Analysis

The forward analysis method predicts future frame processing performance and control voltage/clock. Table 1 describes the parameters which affect to these processing functions. Forward-analysis method predicts the required performance from the following parameters;

1) number of MB block matching: N
2) number of valid coefficients: VC
3) number of valid blocks: VB

The parameters of N, VB, VC are assumed to be predicted from the following equations respectively.

$$N = a \times N' + b \times ABS_f + c \times \Delta Q \tag{5}$$

$$VB = d \times VB' + e \times ABS_f + f \times \Delta Q \tag{6}$$

$$VC = g \times VC' + h \times ABS_f + i \times \Delta Q \tag{7}$$

Where

$$ABS_f: \sum_{l=0,m=0}^{l=176,m=144} |X_{l,m} - Y_{l,m}| \quad \text{@QCIF format}$$

$X_{l,m}$: pixel value of current frame
$Y_{l,m}$: pixel value of previous frame
N': number of MB matching at previous frame
VB': number of valid blocks at previous frame
VC': number of valid coefficients at previous frame
ΔQ: differential of quantization step size between previous frames

To predict N value, three parameters (N', ABS_f and VC) are chosen as affecting parameters;

N': Video sequences have good correlation between frames. When number of MB matching is large in a frame, it is tended to be large in a next frame.
ABS_f: ABS_f indicates the differential between frames, and when ABS_f is large, then N will be large.
ΔQ: The increase of ΔQ results in the prediction error. Prediction error increases the N.

VB and VC are also assumed to be predicted from tree parameters with the same assumption. Furthermore in this paper, required processing performance for motion compensation processing (F_{me}), IQ processing (F_{iq}), IDCT (F_{idct}), VLC (F_{vlc}) are assumed to be predicted from the following equations respectively;

$$F_{me} = j + A \times N \tag{8}$$

$$F_{iq} = k + B \times VC \tag{9}$$

$$F_{idct} = l + C \times VB \tag{10}$$

$$F_{vlc} = m + D \times VC \tag{11}$$

Where

A: processing performance for a MB matching
B: processing performance for a IQ processing
C: processing performance for a IDCT processing
D: processing performance for a VLC processing
j, k, l, m: constant parameter

Total required performance S_p is;

$$F_p = F_{me} + F_{iq} + F_{idct} + F_{vlc} + F_{others} \tag{12}$$

Where

F_{others}: rest of MPEG4 processing

Substituting Eqs. (5)–(11) to Eq. (12), F_p is predicted from the parameters of N', VB', VC', ABS_f and ΔQ, and defined as Eq. (13).

$$\begin{aligned} F_p = {} & n + \alpha \times N' + \beta \times VB' + \gamma \times VC' \\ & + \delta \times ABS_f + \varepsilon \times \Delta Q \end{aligned} \tag{13}$$

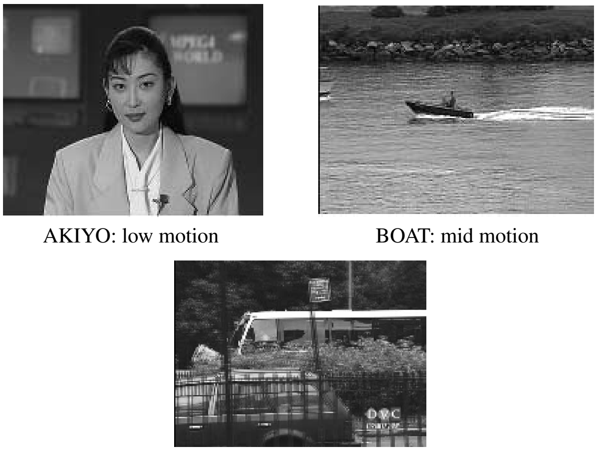
Where
$\alpha, \beta, \gamma, \delta, \varepsilon$: coefficient
n: constant value.

5. Simulation Results

In this section, simulation results are discussed. Section 5.1 describes that TSS with Threshold, which reduce processing power, does not degrade the image quality. Section 5.2 defines the forward-analysis equation, and evaluates the equation from 17 video sequences. The prediction error recovery method at forward-analysis is mentioned at Sect. 5.3.

5.1 TSS with Threshold and TSS without Threshold

At software based MPEG4 codec, TSS with the threshold approach at motion compensation is very useful and well known approach to reduce the processing performance especially at low motion sequences. Here, the simulation results show that TSS with threshold has not worse results objectively than TSS without threshold. Figure 7 shows examples of the video sequences which are used for simulation. The condition of the simulations is described at Table 2.


AKIYO: low motion

BOAT: mid motion

BUS: high motion

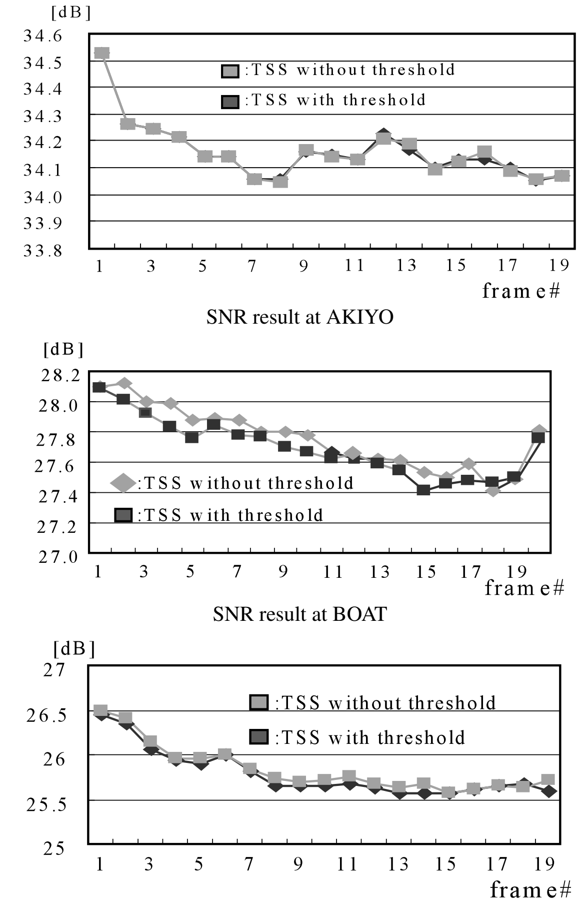
Fig. 7 Examples of evaluated sequences.

Figure 8 is simulation results. At all the video sequences, the TSS with threshold method results in less than 0.1 dB degradation. The degradation can not be objectively recognized. The simulation result definitely shows that reducing processing performance without degrading image can be possible adopting threshold method.

Table 3 indicates the actual numbers of block matching for motion compensation at each sequences. Especially at "Akiyo" sequence, which is low motion sequence, 96.3% of the performance is reduced by introducing threshold method.

Table 2 Simulation conditions.

Frame size	176 by 144 (QCIF)
Frame rate	15 fps
Bitrate	128kbps
Search range	+/- 16 × +/- 16
sequences	Akiyo, boat, bus (shown in Fig.7)



SNR result at AKIYO

SNR result at BOAT

SNR result at BUS

Fig. 8 Simulation result for TSS.

Table 3 Actual numbers of MB matching per frame.

Sequences	TSS without threshold	TSS with threshold	Reduction ratio(%)
Akiyo	5560	202.3	96.3
Boat	5560	1823.8	67.2
Bus	5560	3741.1	32.7

5.2 Clock Frequency Prediction

In order to decide a constant parameters at Eq. (13), simulation has been executed on the reference kit of a 32 bit RISC processor M32R (M32104S6FP) released by Mitsubishi [7] as shown in Fig. 9. The simulation have been led by 17 sequences each of which has originally 150 frames or 5 seconds. The simulation flow is

1) Monitoring actual frequency (F_a), N', VB', VC' and ΔQ from MPEG4 software running on M32R processor
2) Determination of the value of coefficient of n, α,β, γ, δ, and ε of Eq. (13) by the regression analysis method

Also from the simulation results and Eq. (13), predicted frequency F_p (MHz) is defined as;

$$\begin{aligned} F_p\,(\text{MHz}) = {} & (9601460 + 1140.098 \times N' + 2297.982 \\ & \times VB' + 331.4708 \times VC' + 3.400078 \\ & \times ABS_f + 125670.7 \times \Delta Q) \times 15/10^6 \end{aligned} \tag{14}$$

Equation (14) is obtained by the regression analysis method from 1018 points in 17 sequences. Figure 10 shows the correlation between predicted frequency (F_p) from Eq. (14) and actual frequency (F_a). The measured actual frequency lies between 150 MHz and 400 MHz for high quality implementation, depending on characteristics of video sequences. These values are reasonable for the single RISC CPU architecture without additional DSP core [3]. From the Fig. 10, Eq. (14) well predicts the actual required clock-frequency. The case that the predicted frequency is less than the actual frequency results in a failure situation. Therefore, Eq. (14) should be modified in order to avoid the frequent error situation. Prediction mismatch does not occur in the area of $F_p > F_a$ in Fig. 10. By the following Eq. (15) modified from Eq. (14), 99.9% of points satisfy the condition of $F_p > F_a$.

$$\begin{aligned} & F_p\,(\text{MHz}) \\ & = (9601460 + 1140.098 \times N' + 2297.982 \\ & \quad \times VB' + 331.4708 \times VC' + 3.400078 \\ & \quad \times ABS_f + 125670.7 \times \Delta Q) \times 15 \times 1.1/10^6 \end{aligned} \tag{15}$$

5.3 Error Recovery from Prediction Mismatch

The simulation result indicates that the forward analysis method well predicts the actual processing performance and 99.9% frames are predicable from Eq. (14) or Eq. (15). However, there is a possibility that the prediction is mismatched and predicted frequency is smaller than the actual frequency. For this case, we introduce the following processing. For example, as indicated in Fig. 11(a), at every 1/3rd frame, check the completed Macro Block numbers. When completed Macro Block numbers have not been reached to the expected number, voltage/clock frequency is set higher to catch up the processing.

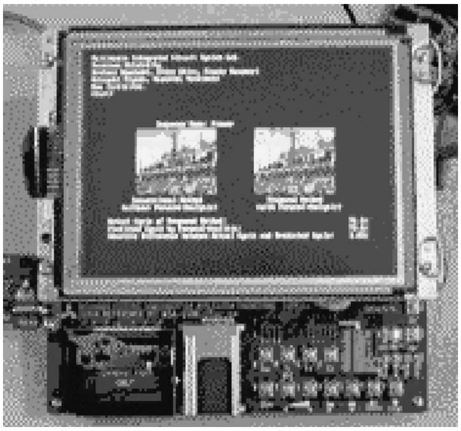


Fig. 9 Simulation system on M32R reference board.

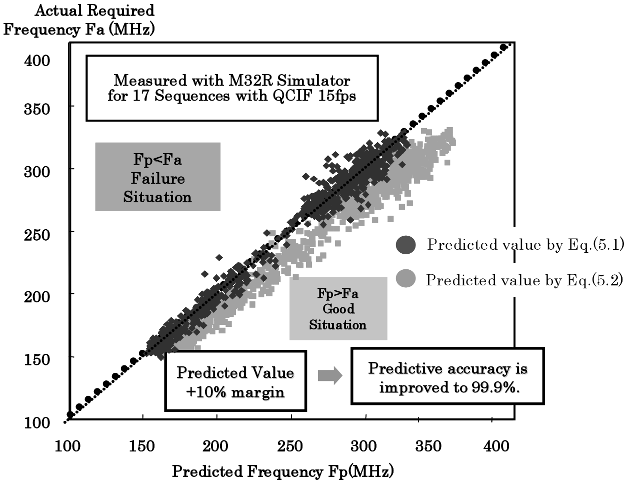
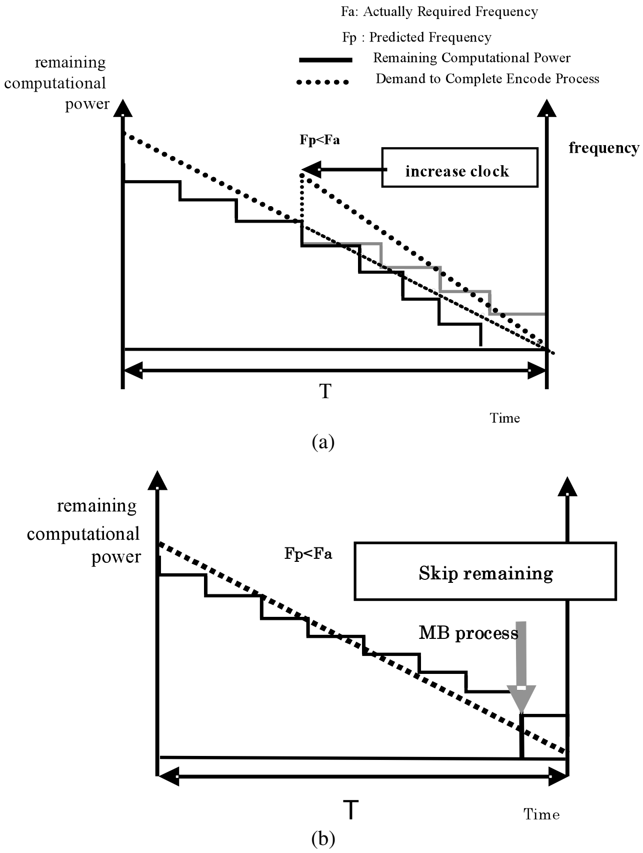


Fig. 10 Predicted frequency (F_p) vs. actual frequency (F_a).

From this error recovery processing, almost all the frames are encoded within the allocated time period. However, even very rare case, the worst case is that the encoding has not been completed at allocated time even after the above mentioned error recovery processing. In this case, unprocessed Macro Blocks (MBs) becomes forcedly not-coded MBs as shown in Fig. 11(b). When MBs become forcedly not coded, the data are compensated from the previous frame. Therefore, these not-coded MBs are recognized as error data, and the quality of the frame is subjectively damaged. At next frame, efficiency of motion compensation becomes worse around “forcedly” not-coded MBs, which may cause some degradation to the next continuous frames. To evaluate both subjective and PSNR quality, we have simulated the case that 3 MBs are forcedly not-coded. From the simulation result, the error frame is subjectively damaged and PSNR is 1.2 dB degradation. However, at the next frame, the error is not subjectively recognized, shown in Fig. 12, and PSNR is less than 0.01 dB degradation. The continuous frames have no degradation in terms of both subjective and PSNR quality. This error recovery processing requires negligible CPU performance.


Fa: Actually Required Frequency
Fp : Predicted Frequency
Remaining Computational Power
Demand to Complete Encode Process
remaining computational power
Fp<Fa
increase clock
frequency
T
Time
remaining computational power
Fp<Fa
Skip remaining
MB process
T
Time

(a)

(b)

Fig. 11 Prediction mismatch error recovery.

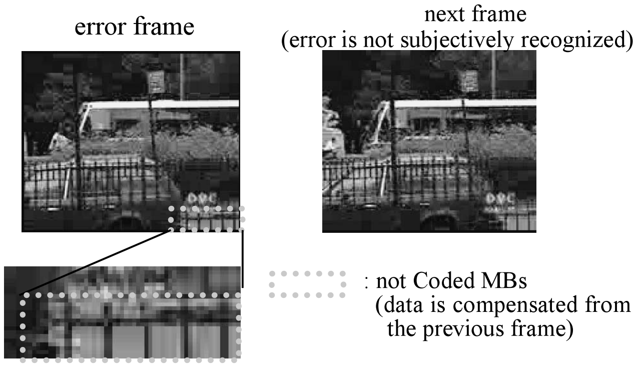

error frame
next frame
(error is not subjectively recognized)
: not Coded MBs
(data is compensated from the previous frame)

Fig. 12 Error frame caused by prediction mismatch.

6. Power Consumption Reduction

Assuming here the f-V characteristics of CPU as Table 4, power consumption of our method (A) and the conventional method (B) has been compared. Power consumption ratio r is defined as follows;

$$r = \frac{P_a}{P_h} = \frac{1}{N}\sum_{i=1}^{N}(p_a/p_h) = \frac{1}{N}\sum_{i=1}^{N}(V_a/V_h)^2 \qquad (16)$$

N: total numbers of frames
P_a: average power consumption of our method (A)
P_h: average power consumption of conventional method (B)
p_a: power consumption per frame controlled by our method
p_h: power consumption per frame at conventional method
V_a: controlled voltage by our method
V_h: conventional voltage (the highest voltage)

The required maximum frequency is roughly 400 MHz assuming the maximum number of the block matching N, the valid block VB and the valid DCT coefficient VC. Therefore, in the Eq. (16), conventional voltage V_h is 1.8 V referring to the Table 4. In the case of low-motion sequence ("AKIYO" sequence), F_p of all the frames are predicted as 150 MHz range, therefore, V_a is 1.0 V. In the case of mid-motion sequences, F_p of sequences is between 243 MHz and 297 MHz, therefore, V_a is between 1.2 V and 1.4 V. At high-motion ("BUS" sequence), V_a is between 1.3 V @270 MHz and 1.7 V @378 MHz. Using Eq. (16), power reduction is estimated. Figure 13 shows the power consumption ratio according to the controllable frequency-voltage (F-V) steps of CPU. The precise F-V control achieves better power consumption reduction especially at high motion sequences, however even fewer F-V steps control can achieves efficient power consumption reduction at low or mid-motion sequences. At nine F-V control steps at Table 4, We achieved 65% reduction at low-motion, 53% reduction at mid-motion, 37% reduction at high-motion. Even using 3 steps F-V set (1.8 V @405 MHz, 1.4 V @297 MHz, 1.0 V @189 MHz), 65% reduction at low-motion, 37% reduction at mid-motion, 36% reduction at high-motion can be achieved.

Table 4 Assumed CPU F-V characteristics.

Frequency (MHz)	f-V steps				Voltage (V)
	2	3	5	9	
405	✓	✓	✓	✓	1.8
378			✓	✓	1.7
351				✓	1.6
324				✓	1.5
297		✓	✓	✓	1.4
270				✓	1.3
243			✓	✓	1.2
216				✓	1.1
189	✓	✓	✓	✓	1.0

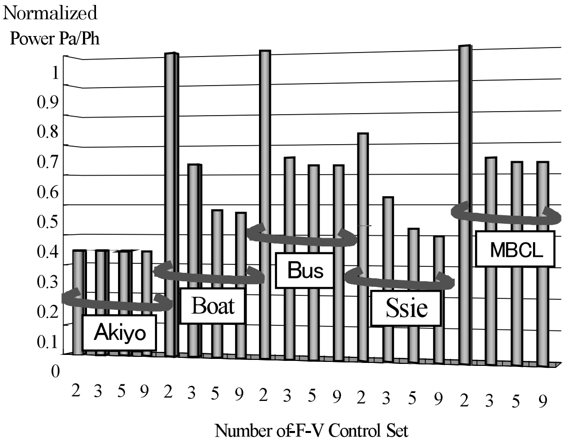


Fig. 13 Normalized power ratio according to the numbers of f-V steps.

7. Conclusions

Low power approach for MPEG4 encoding processing on multi-regulated voltage CPU has been presented. Combining the recent low power and high performance CPU technology and characteristics of video compression processing, the proposed method achieves theoretically minimized power consumption by feed-forward dynamic voltage/clock-frequency control according to the predicted processing performance. Simulation results indicate that feed-forward analysis well predicts the actual processing performance. By controlling clock-frequency/Voltage of CPU dynamically by every frame, approximately 65% power consumption reduction at low motion sequence, 53% reduction at mid motion sequence, 37% reduction at high motion sequence can be achieved.

References

[1] M. Takahashi, M. Hamada, T. Nishikawa, H. Arakida, Y. Tsuboi, T. Fujita, F. Hatori, S. Mita, K. Suzuki, A. Chiba, T. Terazawa, F. Sano, Y. Watanabe, H. Momose, K. Usami, M. Igarashi, T. Ishikawa, M. Kanazawa, T. Kuroda, and T. Furuyama, "A 60 mW MPEG4 video codec using clustered voltage scaling with variable supply-voltage scheme," Proc. ISSCC1998, pp.36–37, 1998.

[2] H. Ohira, T. Kamemaru, H. Suzuki, K. Asano, and M. Yoshimoto, "A low power media processor core performable CIF30fr/s MPEG4/H26x video codec," IEICE Trans. Electron., vol.E84-C, no.2, pp.157–165, Feb. 2001.

[3] T. Yamada, M. Ishikawa, Y. Ogawa, T. Tsunoda, T. Irita, S. Tamaki, K. Nishiyama, T. Kamei, K. Tatezawa, F. Arakawa, T. Nakazawa, T. Hattori, and K. Uchiyama, "Low-power embedded risc microprocessor with an integrated DSP for mobile applications," IEICE Trans. Electron., vol.E85-C, no.2, pp.253–262, Feb. 2002.

[4] J. Heeb, "Next generation Intel StrongARM technology overview," CoolChips2000, pp.137–148, April 2000.

[5] H. Yasuura and T. Ishihara, "System LSI design methods for low power LSIs," IEICE Trans. Electron., vol.E83-C, no.2, pp.143–152, Feb. 2000.

[6] H. Kawaguchi, G. Zhang, S. Lee, and T. Sakurai, "An LSI for VDD-hopping and MPEG4 system based on the chip," Proc. IEEE International Symposium on Circuits and Systems, pp.918–921, May 2001.

[7] Mitsubishi 32-bit RISC Single-chip Microcomputers M32R Family M32R/ECU Series32170 32174 Group User's Manual.

[8] K. Kawakami, H. Ohira, M. Kanamori, M. Miyama, and M. Yoshimoto, "A feed-forward dynamic voltage control algorithm for low power/high quality MPEG4 on multi-regulated voltage CPU," CoolChips2003, pp.87–101, April 2003.

Hideo Ohira received the B.S. degree in mechanical engineering from Tokyo University, Japan in 1986. During 1986–2000, he had researched digital image signal processing and video LSI architecture at Mitsubishi Electric Corp. in Japan. At 1993–1994, he was a visitor researcher at University California, Davis in US, where he has researched very high resolution image processing. At 2000, he joined Packetvideo corp., where he had been engaged in deploying video-streaming platform for wireless terminals. At 2002, he joined NTTdocomo, where he was a team manager of video streaming services for 3G mobile phone. Currently he is working at Nvidia corporation in Japan, and also in the Ph.D. course at Graduate School of Natural Science and Technology, Kanazawa University in Japan. Mr. Ohira is a member of the Institute of Image Information and Television Engineers (IITE).

Kentaro Kawakami received the B.S. degree in electrical and information engineering from Kanazawa University, Ishikawa, Japan, in 2002. He is currently a master's student at Graduate School of Natural Science and Technology, Kanazawa University. His current research interests are RISC based image codec VLSI.

Miwako Kanamori received B.S. degree in electrical and information engineering from Kanazawa University, Ishikawa, Japan, in 2003. She is currently a master's student at Graduate School of Natural Science and Technology, Kanazawa University. Her current research interests are RISC based image codec VLSI.

Yasuhiro Morita entered electrical and electronic Engineering, Kanazawa University in 2000. In 2003, he skipped one year upon completing 3rd year of university, and he is currently master's student at Graduate School of Natural Science and Technology, Kanazawa University.

Masayuki Miyama received the B.S. degree in computer science from University of Tsukuba in 1988. He Joined PFU Ltd. in 1988. He received the M.S. degree in computer science from Japan Advanced Institute of Science and Technology in 1995. He joined Innotech Co. in 1996. He is a research assistant in the Department of Electrical and Electronic Engineering at Kanazawa University. His present research focus is low power design technique for multimedia VLSI.

Masahiko Yoshimoto received the B.S. degree in electronic engineering from Nagoya Institute of Technology, Nagoya, Japan, in 1975, and M.S. degree in electronic engineering from Nagoya University, Nagoya, Japan, in 1977. He received Ph.D. degrees in Electrical Engineering from Nagoya University, Nagoya, Japan in 1998. He joined the LSI Laboratory, Mitsubishi Electric Corporation, Itami, Japan, in April 1977. From 1978 to1983 he had been engaged in the design of NMOS and CMOS static RAM including a 64K full CMOS RAM with divided-word-line structure. Since 1984 he had been involved in the research and development of a digital NTSC decoder LSI with adaptive filtering called VSP, an image compression DSP called DISP, a 100 MHz DCT processor, MPEG2 video encoder/decoder LSIs, 3-D graphics processor, multimedia ULSI systems for the digital broadcasting and the digital communication systems based on MPEG2 and MPEG4 Codec LSI core technology and so on. Since 2000, he has been a professor of Dept. of Electrical & Electronic System Engineering in Kanazawa University, Japan. His current activity is focused on the research and development of multimedia and ubiquitous media VLSI systems including an ultra low power image compression processor and a low power wireless interface circuit. He holds on 70 registered patents. He has served on the program committee of the IEEE International Solid State Circuit Conference from 1991 to 1993. Also he served as Guest Editor for special issues on Low-Power System LSI, IP and Related Technologies of IEICE Transactions in 2004. He received the R&D100 awards from the R&D magazine for the development of the DISP and the development of the real-time MPEG2 video encoder chipset in 1990 and 1996, respectively.