行政院國家科學委員會專題研究計畫　期中進度報告

子計畫六：用於奈米晶片系統設計之功率意識高階合成研究

(1/3)

計畫類別： 整合型計畫
計畫編號： NSC92-2220-E-009-033-
執行期間： 92年11月01日至93年07月31日
執行單位： 國立交通大學電機與控制工程學系

計畫主持人： 董蘭榮

計畫參與人員： 楊學之，顏介皇，張騰轟，連樹德，簡芳彥，葉松樹，楊明峰，陳育聖，葉丁豪

報告類型： 完整報告

處理方式： 本計畫涉及專利或其他智慧財產權，1年後可公開查詢

中　華　民　國 93年5月31日

中英文摘要

（一） 計畫中文摘要

在 CMOS 電路尺寸逼近極限時，電路的功率消耗與效能品質的平衡考量就顯得非常重要。特別是隨著奈米技術的大幅進展，能意識到耗能條件的設計方法成為很重要的關鍵。所謂功率意識設計除了考慮平均的功率消耗外同時也包含了瞬間的功率消耗狀態，例如：峰值功率，功率梯度等。目前考慮瞬間功率消耗的設計方法僅限於電晶體或是邏輯閘階層的設計。然而，如果在系統設計階段就功率意識找出解答，積體電路的設計將可明顯地提升低階技術的功率意識最佳化程度。本計畫的目的就在於探索可用於功率意識系統的高階合成方法來管理並減少暫態功率。

關鍵詞: 功率意識；高階合成；系統晶片；電腦輔助設計

（二） 計畫英文摘要

As we get closer to the limits of scaling in CMOS circuits, it is imperative to consider power/performance trade-offs and to develop appropriate power aware methodologies and techniques for embedded systems. The use of nanometer technologies is making it increasingly important to consider transient characteristics of a circuit's power dissipation (e.g., peak power, and power gradient or differential) in addition to its average power consumption. State-of-the-art transient power analysis and reduction approaches are mainly at the transistor- and gate-levels. However, we believe architectural solutions to transient power problems may complement and significantly extend the scope of lower-level techniques, as was the case with average power minimization. This project intends to exploit high-level synthesis approach to transient power management and reduction in that a power-aware high-level synthesis can impact the cycle-by-cycle peak power and peak power differential for the synthesized implementation.

Keywords: Power-aware system; High-level synthesis; SOC; CAD

目錄

報告內容

一、 前言

With increasing demand of portable, power-aware multimedia devices, an architecture that can be flexible in both power consumption and performance is highly required. As we get closer to the limits of scaling in CMOS circuits, it is imperative to consider power/performance trade-offs and to develop appropriate power aware methodologies and techniques for embedded systems. The use of nanometer technologies is making it increasingly important to consider transient characteristics of a circuit's power dissipation (e.g., peak power, and power gradient or differential) in addition to its average power consumption. State-of-the-art transient power analysis and reduction approaches are mainly at the transistor- and gate-levels. However, we believe architectural solutions to transient power problems may complement and significantly extend the scope of lower-level techniques, as was the case with average power minimization.

When circuit technology scales through shrinking the transistor feature size by a factor of x, the capacitance is reduced by x and the supply voltage by x^2. Therefore, power decreases by a factor of x^3, provided the frequency remains the same. Unfortunately, with each generational scaling of the feature size, more complex, aggressive designs are used. These designs employ higher clock frequency, larger chip area and higher total number of transistors due to the use of more aggressive speculative execution. The result is a significant increase in power dissipation. On the other hand, aggressive, complex designs increase the opportunities available fore power management: there are more individual units which can be placed on standby when not needed by the application.

Another worrying trend is the increase in power density. Considering the Intel family of microprocessors, for instance, the power density is expressed in terms of watts/cm^2 : the current generation is getting close to the power density of a nuclear reactor. This results in more expensive cooling mechanisms and reduced reliability. The increase in total power dissipation as well as power density means that traditional power management policies centered only at the device and VLSI levels are no longer sufficient. As a result, power has propagated as an important design constraint to the higher levels. Therefore, this project intends to exploit high-level synthesis approach to transient power management and reduction in that a power-aware high-level synthesis can impact the cycle-by-cycle peak power and peak power differential for the synthesized implementation.

二、 研究目的

As mentioned above, with increasing demand of portable, power-aware multimedia devices, an architecture that can be flexible in both power consumption and performance is highly required. This project will first investigate and characterize power consumption of battery components and then come up with high level synthesis approaches to balance the power dissipation and performance and thus save the power consumption while maintaining required system performance. Given the transient power constraints, the proposed project has four goals: to have the longest battery lifetime while achieving the performance goals, to deliver task schedule and resource allocation automatically, and to synthesize the SOC architectures at system level.

三、 文獻探討

Currently, power-aware systems research at the architectural level for power saving is concentrated on the following issues: instruction set architecture (ISA) selection, instruction caches (I-cache) and the system bus, voltage and frequency scaling, battery-consciousness, and task movement.

1. **Instruction Set Architecture (ISA) Level**: This is an active research area in the context of general-purpose architectures; various researchers have commented on the need to take power and energy into account in ISA design. However, not much effort has been devoted to power-aware ISA design. Paper [1] employs a fine-grained off-line scheduling approach which saves power by combining multiple instructions into on complex but lower power instruction or by using low-power versions of instructions while considering task deadlines. The proposed scheme assumes that the ISA is sufficiently flexible; however, in practice there is not much scope for the existence of complex instructions which are functionally equivalent to a group of simpler instructions in the ISA design.
2. **I-cache and Buses**: The control path, which governs the fetch, issue and retiring of instructions, is quite simple in typical embedded processors and occupies a relatively small portion of the chip area. The caches take up most of the chip area [2] and are responsible for a considerable percentage of the energy dissipation even though memory is more energy efficient than control logic. Paper [3] compresses the instructions in memory. This saves instruction fetch energy by using fewer bits on a fetch. An alternative strategy by paper [4] also concentrates on saving instruction energy. The authors employ a loop cache and keep the tight loop in a small loop cache instead of accessing a larger block.

This paper shows the usefulness of augmenting an ISA in a power-aware fashion.

3. **Voltage and Frequency Scaling**: In general, complex systems are typically over-designed, provisioning resources for the worst-case execution time. Since tasks rarely execute up to their worst case, there is significant scope for power and energy savings using dynamic voltage and frequency scaling. Papers [5]-[8] are in this category.
4. **Battery Consciousness**: The most important issues to be considered for battery-driven systems are the total battery capacity and the battery discharge profile. The latter is important in devising battery-aware schemes that are guided by the discharge profile. Paper [9] considers distributed real-time systems and develop battery model, which is used in two scheduling schemes: first they optimize the battery discharge power profile, and then they use voltage scaling for distributed real-time systems. The overall objective is to extend the battery lifespan while meeting task deadlines and precedence requirements. The authors claim that mitigating battery capacity loss requires reducing the discharge current level and shaping its distribution.
5. **Task Movement**: Task movement is important in real-time systems for fault-tolerance or load balancing purposes. However, power efficient task movement heuristics have not been extensively investigated. One exception is the work of paper [10]. The paper is based on the observation that a set of processors can operate at a lower power level than a single one with the same performance if there is enough parallelism.

四、 研究方法

We consider the project as three parts: transient power management thru high-level synthesis, system-level power-aware design automation, and high-level synthesis for adaptive power-quality tradeoff in energy-aware multimedia embedded systems. The yearly schedule is shown as follows:

1st Year:

1. Study on power characteristics of battery-based system.
2. Develop static scheduling algorithm under transient power constraints.
3. Demonstrate the proposed scheduling technique using state-of-the-art commercial design flow.

2nd Year:

1. Study on power aware Instruction Set Architecture (ISA).
2. Develop automated ISA selection for power aware systems.

3. Develop power-aware bus encoding techniques.
4. Develop the power-aware scheduling algorithm for dynamic voltage and frequency scaling.

3rd Year:

1. Study on battery-conscious multimedia systems.
2. Develop the adaptive power-quality tradeoff algorithm.
3. Develop the high-level synthesis for adaptive power management.

By the end of this project, we would expect as follows:

1. Publications: There will be at least two papers published in major international conferences each year. We will publish at least two academic journal papers in support of this three-year project. Also, there will be at least two Ph.D. dissertations and six master theses funded by the project.
2. CAD environment: We are going to build up a high-level synthesis tool driven by techniques from this project and embedded the tool into state-of-the-art commercial design flow.
3. Training: There will be two Ph.D. students and six master students earned their degrees within the execution period of this project.

五、 結果與討論

The project has resulted in two journal papers and a conference paper:

1. Hsien-Wen Cheng and Lan-Rong Dung, "A Vario-Power Motion Estimation Architecture Using Content-based Subsample Algorithm," IEEE transactions on Consumer Electronics, Feb. 2004, pp.349-354.
2. Hsien-Wen Cheng and Lan-Rong Dung, "A Power-Aware ME Architecture Using Subsample Algorithm," ISCAS 2004.
3. Lan-Rong Dung and Hsueh-Chih Yang, "On Multiple-Voltage High-Level Synthesis Using Algorithmic Transformations," conditional accepted by IEICE Trans. Fundamentals, 2004.

The following pages are the articles. The first two articles are published and the last one is on revision phase.

A Vario-Power ME Architecture Using Content-Based Subsample Algorithm

Hsien-Wen Cheng and Lan-Rong Dung, Member, IEEE

Abstract —*The Motion estimator is a key element in many video compression systems and it tends to dominate the power consumption in them. With increasing demand of portable, power-aware multimedia devices, an architecture that can be flexible in both power consumption and compression quality is essential. To meet this requirement, this paper presents a novel power-aware architecture, called the Vario-Power Architecture, for the motion estimation. Based on a semi-systolic array with the content-based subsample algorithm, the architecture real-time disables some processing elements to reduce power consumption. By performing the edge extraction first, a threshold is then set as the criterion of whether to enable or disable processing elements and thus the switch activities of the system can be reduced. As the simulation shows, the architecture may operate at different power consumption modes according to the remaining capacity of the battery pack giving little quality degradation and the power overhead under 0.36%.*[1]

Index Terms —motion estimation, VLSI architecture, video compression, power-aware architecture, subsample.

I. Introduction

The technique of motion estimation has been widely used in the video compression system for years. As the demand of portable, power-aware multimedia devices increases, an architecture that can be flexible in both power consumption and compression quality is essential. This paper presents a power-aware architecture, called the Vario-Power Architecture, which provides real-time switching of power consumption mode for conserving battery life[2] while it maintains the compression quality in high level. The proposed architecture is driven by the content-based subsample algorithm which maintain acceptable quality degradation when the system works at different power consumption modes; that is, the proposed architecture can perform trade-offs between power consumption and compression quality as the battery status changes. Because the control mechanism and data sequences at different power consumption modes are the same, the vario-power architecture can perform the switching of power consumption mode very smoothly. The block diagram shown in Fig.1 illustrates a typical application of the vario-power architecture for motion estimation in a video compression system. The host processor monitors the remaining capacity of the battery pack and switches the operation mode of the vario-power element to maintain the performance of the compression system better.

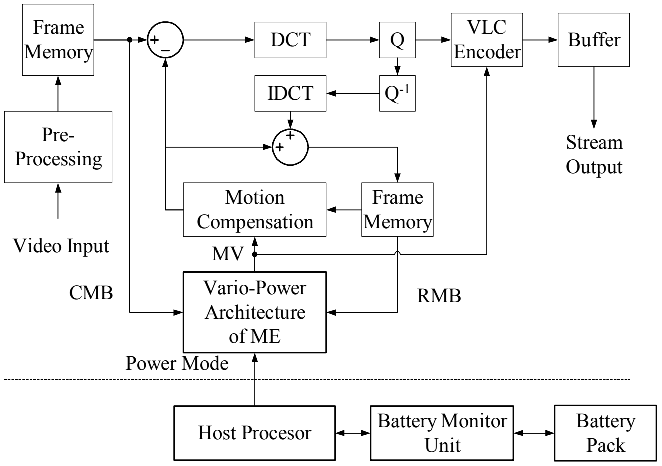


Fig. 1. A typical application of the vario-power ME architecture in a video compression system.

Lots of published papers presented efficient algorithms for the VLSI implementation of motion estimator either on high performance or low power design. Yet, most proposed algorithms cannot adapt their system to different power consumption modes. Among these proposed algorithms, the Full-Search Block-matching (FSBM) algorithm with Sum of Absolute Difference (SAD) criteria is the most popular approach for motion estimation because of its considerably good quality. There are many types of architecture which have been proposed for the implementation of FSBM algorithms [2] [3] [4] [5]. However, a huge number of comparison/difference operations result in high computational load and significant power consumption. To reduce the computational complexity of FSBM, researchers have proposed various fast algorithms by reducing the searching steps [6] [7] [8] [9] [10]. Unfortunately, these fast algorithms suffer from irregular block-matching scheme and much worse quality than FSBM, and they are not suitable for the implementation of vario-power architecture. Papers in [11] and [12] may reduce the computational load without degrading the compression quality; nevertheless, they both require additional operations for each search step and cannot adapt themselves to different power consumption modes. The Novel Early-Jump-Out (NEJO) technique in [13] addressed the low-power architecture with some quality degradation. However, the EJO also requires extra operations for each search step and is not feasible enough for vario-power architecture.

[1] This work was supported in part by Taiwan MOE Program for Promoting Academic Excellent of Universities under the grant number 91-E-FA06-4-4 and the National Science Council, R.O.C., under the grant number NSC 92-2220-E-009-003.

The authors are with the Department of Electrical and Control Engineering, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 300, Taiwan, R.O.C. (email: jswcheng.ece89g@nctu.edu.tw; lennon@cn.nctu.edu.tw).

[2] The idea is motivated from the SpeedStep$^{\text{TM}}$ technology of Intel [1].

Contributed Paper
Original manuscript received July 11, 2003
Revised manuscript received January 29, 2003

0098 3063/04/$20.00 © 2004 IEEE

Articles in [14] and [15] present subsample algorithms to significantly reduce the computation cost with low quality degradation. The reduction of computational cost implies saving of the power consumption, and the power consumption can be reduced by simply increasing the subsample rate; thus, the subsample algorithms are very suitable for implementing the vario-power architecture. However, applying the subsample algorithms for the vario-power architecture may suffer from aliasing problem for high frequency band. The aliasing problem makes compression quality degrading rapidly as the subsample rate increases. To alleviate the problem, we extend the traditional subsample algorithm to a novel algorithm, which is called the content-based subsample algorithm. In our approach, we first use edge extraction techniques to separate the high-frequency band from a macro-block and then perform subsampling within the low-frequency band only. Based on the architecture proposed by Hsieh and Lin [4], we present a semi-systolic architecture which is driven by the content-based subsample algorithm. The proposed architecture can real-time alter the subsample rate as the power consumption mode changes. As the result shows, our methodology successfully switches the power mode while the quality degradation is a little.

II. General subsample algorithm

The FSBM algorithm with SAD criteria is the most popular approach for motion estimation because of its considerably good quality and many published papers presented efficient architectures for the VLSI implementation of it [2] [3] [4] [5]. To determine the motion vector, the algorithm uses (1) and (2) to compare each current macro-block (CMB) of the frame with all the reference macro-blocks (RMB) in searching area.

$$SAD(u,v)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left|S(i+u,j+v)-R(i,j)\right|, \quad \text{where } -p\le u,v\le p-1. \tag{1}$$

$$SAD_{\overline{MV}}=\min\nolimits_{-p\le u,v\le p-1} SAD(u,v)$$
$$and\ \overrightarrow{MV}=(u,v)\Big|_{\min_{-p\le u,v\le p-1} SAD(u,v)} \tag{2}$$

where the macro-block size is *N-by-N*, *R(i,j)* is the luminance value at *(i,j)* in the current macro-block (CMB). The *S(i+u,j+v)* is the luminance value at *(i,j)* of the reference macro-block (RMB) which is offset *(u,v)* from the CMB in the searching area which size is *2p-by-2p*.

Many researches addressed subsample techniques for motion estimation to reduce the computation load of FSBM [14] [15]. Liu and Zaccarin, as pioneers of the subsample algorithms, significantly reduced the computational load by applying the *4-to-1* alternative subsampling to FSBM. As the simulation results show, the subsample algorithm reduces the computational load by a factor of 8 or 16 while the quality is similar to that of exhaustive search [14]. Here, we present the general subsample algorithm in which the subsample rate ranges from *4-to-1* to *1-to-1* for the vario-power architecture. The general subsample algorithm shown in (3) uses the matching criterion which is called subsample sum of absolute difference (SSAD). In (3), the $SM_{8:m}$ which is generated from (4) is the subsample mask for the subsample rate *8-to-m*.

$$SSAD_{8:m}(u,v)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left|SM_{8:m}(i,j)\cdot\left(S(i+u,j+v)-R(i,j)\right)\right|$$
$$\text{for - } p\le u,v\le p\text{ - }1, \tag{3}$$
$$\text{where } SM_{8:m}(i,j) \text{ is the mask for } 8\text{ - }to\text{ - }m \text{ subsample and}$$
$$SM_{8:m}(i,j)=BM_{8:m}(i \bmod 4, j \bmod 4)$$

$$BM_{8:m}=\begin{bmatrix} u(m-2) & u(m-5) & u(m-2) & u(m-6)\\ u(m-3) & u(m-7) & u(m-4) & u(m-8)\\ u(m-2) & u(m-5) & u(m-2) & u(m-6)\\ u(m-3) & u(m-7) & u(m-4) & u(m-8) \end{bmatrix} \tag{4}$$
$$\text{where } u(n) \text{ is a step function; that is, } u(n)=\begin{cases}1, & for\ n\ge 0\\ 0, & for\ n<0\end{cases}.$$

Using the general subsample algorithm, the power consumption can be reduced by simply increasing the subsample rate in which way the reduction of computation cost implies the saving of power consumption. Obviously, the general subsample algorithm is very suitable to implement the vario-power architecture. However, the algorithm is rather content independent and suffers from aliasing problem in high frequency band. To enhance the performance, we introduce a content-dependent technique to the general subsample algorithm as addressed in the following section.

III. Content-Based Subsample Algorithm

As mentioned above, the general subsample algorithm has aliasing problem when it is in high subsample rate. The aliasing problem leads to considerable quality degradation because the high frequency band is messed up. To alleviate the problem, we use edge extraction techniques to separate the edge pixels from a macro-block and then perform subsampling to the remaining pixels.

Fig. 2 describes the procedure of the content-based subsample algorithm. The algorithm first determines edge pixels of the current macro-block and then generates the content-based subsample mask (CSM). Upon the CSM generated, we are able to calculate SSAD values and find the best motion vector. The determination of edge pixels starts from applying gradient filter in the current macro-block. In this paper, we use three popular gradient filters which are the high-pass filter, the Sobel filter and the morphological gradient filter to exercise the performance of the content-based algorithm [16]. Equations (5)-(8) illustrate their gradient calculations.

```
Input current and reference $W \times H$ frames;
for($y = 0; y < W/N; y++$){
  for($x = 0; x < H/N; x++$){
    Perform gradient filtering;
    Calculate the edge threshold accroding to power mode;
    Determine edge pixels and edge mask;
    Generate content-based subsample mask (CSM);
    $SSAD_{\min}(x, y) = \infty$;
    for($u = -p; u < p; u++$){
      for($v = -p; v < p; v++$){
        $SSAD(u,v) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left|CSM(i,j)\cdot\left(S(i+u,j+v)-R(i,j)\right)\right|$;
        if $SSAD_{\min}(x,y) > SSAD(u,v)$
          {$SSAD_{\min}(x,y) = SSAD(u,v); MV(x,y) = (u,v);$ }
      }//for loop index $v$
    }//for loop index $u$
  }//for loop index $x$
}//for loop index $y$
```

Fig. 2. The procedure of the content-based subsample algorithm.

High-Pass Filter:

$$G_{hpf}(i,j) = \left|MF\left(HPF_{mask}, R(i,j)\right)\right|,$$
$$\text{where } HPF_{mask} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}. \tag{5}$$

Sobel Filter:

$$G_{sobel}(i,j) = \left|MF(SX_{mask}, R)(i,j)\right| + \left|MF(SY_{mask}, R)(i,j)\right|,$$
$$\text{where } SX_{mask} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}, SY_{mask} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{6}$$

In (5) and (6), the *MF(·)* function is the mask filter operation which is shown in (7).

$$MF\left(M, R(i,j)\right) = \sum_{p=-1}^{1}\sum_{q=-1}^{1} M(p+1, q+1)\cdot R(i+p, j+q),$$
$$\text{where } M \text{ is a } 3-by-3 \text{ mask and } R(i,j) \text{ is the luminance value at } (i,j) \tag{7}$$

Morphological Gradient filter:

$$G_{morphological} = (R \oplus B) - (R \Theta B),$$
$$\text{where } B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{8}$$

The operation '$\oplus$' and 'Θ' denote the morphological operation of dilation and erosion respectively.

After obtaining the gradient value *G*, we first determine the edge threshold of this CMB as defined in (9).

$$threshold = m_1 \cdot \max\{G(i,j)\} + m_2 \cdot \min\{G(i,j)\},$$
$$\text{where } m_1 \text{ and } m_2 \text{ are set according to power mode.} \tag{9}$$

Then, the algorithm uses the threshold value as a condition to pick the edge pixels and produces the edge mask by (10).

$$EdgeMask(i,j) = \begin{cases} 1, & for\ G(i,j) \geq threshold \\ 0, & otherwise \end{cases} \tag{10}$$

Finally, the contend-based subsample mask (CSM) is generated from (11) and, therefore, the content-based subsample rate (CSR) is N^2-to-(the number of 1's in CSM).

$$CSM(i,j) = SM_{8:m}(i,j)\ OR\ EdgeMask(i,j)$$
$$\text{for } 0 \leq i,j \leq N-1. \tag{11}$$

Since the higher the threshold value is, the less the edge pixels will be, the CSM is highly dependent on the threshold. Thus, the switching of power mode can be done by adjusting the threshold parameters *m1* and *m2*.

IV. The Vario-Power Architecture

According to the content-based subsample algorithm, we present a semi-systolic architecture which is based on the architecture proposed by Hsieh and Lin [4]. The vario-power architecture shown in Fig. 3 contains an edge-extraction unit (EXU), an array of processing elements (PEs), a parallel adder tree (PAT), a shift register array (SRA), and a motion-vector selector (MVS). Given the power consumption mode, the EXU extracts high frequency pixels (or edge pixels) from the current macro-block (CMB) and generates *0-1* content-based subsample masks (CSM) for the PE array to disable or enable processing elements (PEs). As shown in Fig. 4, the structure of PE array, which is used to accumulate the absolute differences of pixels column by column while the parallel adder tree sum up all the results to generate the subsample sum of absolute difference (SSAD). Finally, The MVS performs compare-and-select operation to select the best motion vector which has a minimum SSAD.

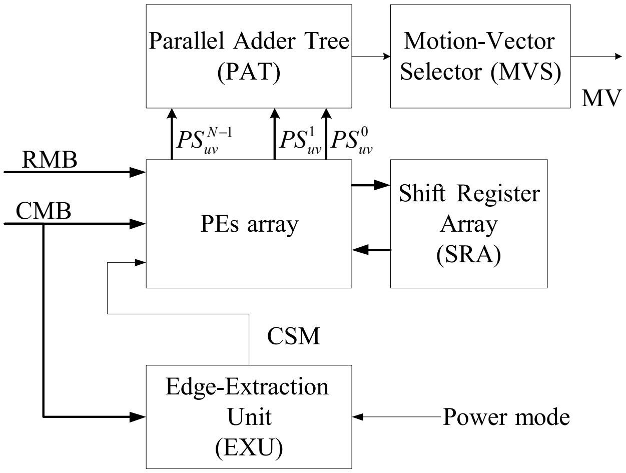

Parallel Adder Tree (PAT)
Motion-Vector Selector (MVS)
MV
PS_{uv}^{N-1}
PS_{uv}^{1}
PS_{uv}^{0}
RMB
CMB
PEs array
Shift Register Array (SRA)
CSM
Edge-Extraction Unit (EXU)
Power mode

Fig. 3. The system block diagram of the content-based subsample algorithm.

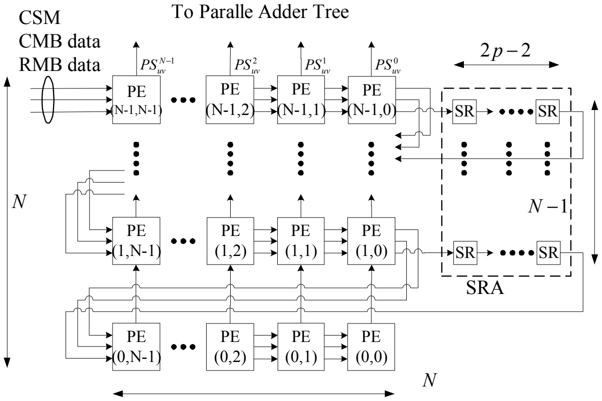


Fig. 4. The architecture of PE array and shift register array.

Since the adder operations which PE achieves dominant significant power consumption in the system, the proposed architecture driven by the content-based subsample algorithm can disable some processing elements to save the switch activity of PE; that is, it can save the power consumption of the system. By performing the edge extraction first, a threshold was then set as the criterion of how many processing elements will be turn-off. Figure 5 shows the structure of the Processing Element and explains how the CSM disables or enables the PE. The absolute difference (AD) unit, which is denoted as |*a*-*b*|, calculates the absolute difference of CMB and RMB pixels. Then the adder unit accumulates the absolute difference value with the partial sum from the previous PE and conveys the results to the next PE in the same column. In order to reduce the switch activity, the PE receives the CSM signals from the EXU to disable or enable the blocking registers, which is abbreviated as 'breg' in Fig. 5, to decide the PE is active or inactive. When the blocking registers are enabled, the data paths of AD unit and the adder unit remains still, that is, the switch activity of this inactive PE is reduced. Thus, the power consumption can be saved.

The edge-extraction unit contains two main blocks which are gradient filter and CSM generator. The implementation of gradient filter is based on one of the (5), (6) and (8). Figure 6, for instance, is the structure of the implementation for the high-pass gradient filter. The multiplexers are used to prevent the boundary error for border pixels of the CMB. The black-dot in each multiplexer indicates the switching path when the filter is processing a border pixel. The CSM generator, whose structure is illustrated in Fig. 7, figures out the maximum and minimum of the gradient value in the macro-block and then determines the threshold value according to the power mode as shown in (9). Finally, it generates the CSM by OR-merging the regular subsample pattern and the edge pattern.

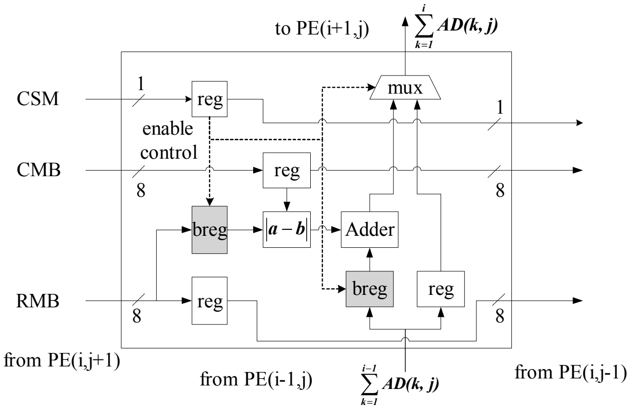


Fig. 5. The PE structure for the vario-power architecture.

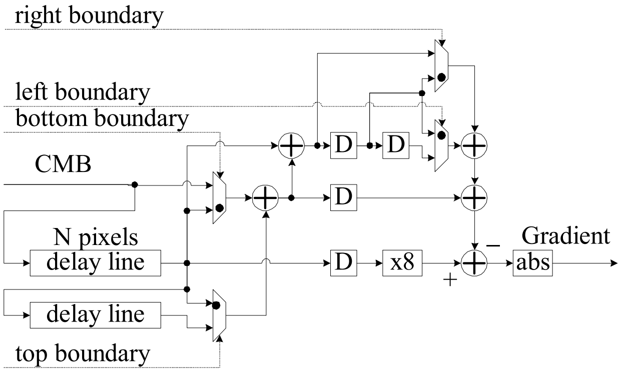


Fig. 6. The architecture to implement the high-pass gradient filter.

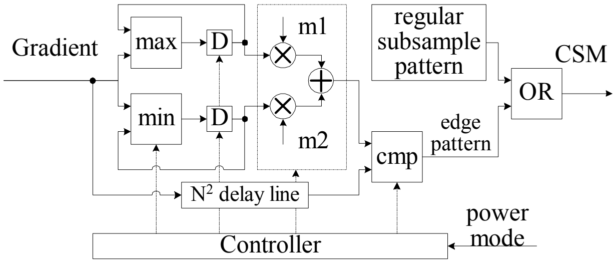


Fig. 7. The architecture of the CSM generator.

The execution of the vario-power architecture has five phases: initial CMB phase, filtering phase, edge-determination phase, initial RMB phase, and SSAD calculation phase. The initial CMB phase is for loading the CMB data into PE array and the initial RMB phase is for filling up PE array in full with RMB data to start the SSAD calculation. Figure 8 illustrates the timing of data flow. Since the execution of edge-extraction

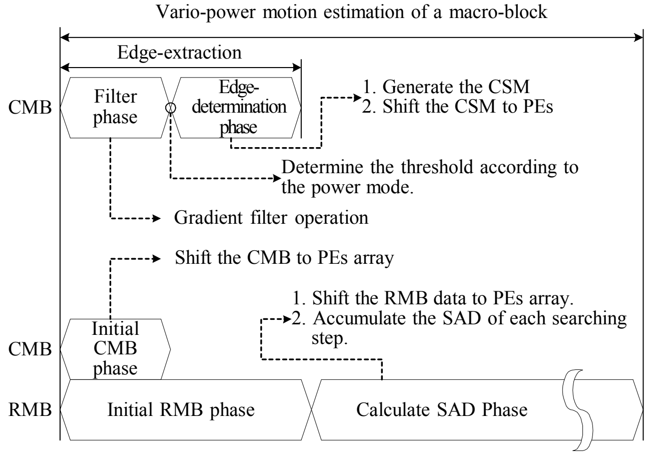

Vario-power motion estimation of a macro-block
Edge-extraction
CMB
Filter phase
Edge-determination phase
1. Generate the CSM
2. Shift the CSM to PEs
Determine the threshold according to the power mode.
Gradient filter operation
Shift the CMB to PEs array
1. Shift the RMB data to PEs array.
2. Accumulate the SAD of each searching step.
CMB
Initial CMB phase
RMB
Initial RMB phase
Calculate SAD Phase

Fig. 8. The data flow of the vario-power architecture.

unit, which includes filtering phase and edge-determination phase, is parallel to the initial RMB phase, the behavior of the proposed architecture is the same as the architecture without vario-power function proposed by Hsieh and Lin [4].

V. Estimation of Power Consumption

From (12), the power consumption of digital VLSI is in proportion to the switch activity, $f \cdot \gamma(0 \leftrightarrow 1)$.

$$P_{gate} = f \cdot C_{gate} \cdot V_{DD}^2 \cdot \gamma(0 \leftrightarrow 1), \qquad (12)$$
$$\text{where } \gamma(0 \leftrightarrow 1) \text{ is the switching rate of the gate.}$$

Because the addition is the majority of the motion estimation architecture, by referring to [11], this paper uses the number of equivalent additions, denoted as ε_{adder}, as the power measure unit to estimate the power consumption. As per Fig. 5, the calculation of an absolute difference nearly requires $2\varepsilon_{adder}$ and each PE consumes $3\varepsilon_{adder}$ in each iteration. When the PE array operates at the content-based subsample rate of R_s, it requires $R_s^{-1} \cdot N^2 \cdot (2p)^2 \cdot 3\varepsilon_{adder}$. Since the calculation of PAT requires $(N-1) \cdot (2p)^2 \cdot \varepsilon_{adder}$, the power consumption of calculating SSADs for all RMBs in searching area is $12R_s^{-1}N^2p^2\varepsilon_{adder} + 4(N-1)p^2\varepsilon_{adder}$.

As regard to the EXU, three gradient filters which are mentioned in this paper consume $6N^2\varepsilon_{adder}$, $9N^2\varepsilon_{adder}$ and $8N^2\varepsilon_{adder}$,respectively, and the edge-determination consumes $3N^2\varepsilon_{adder}$. So the power consumption of the content-based subsample algorithm (P_{CSA}) can be expressed as (13).

$$P_{CSA} \cong 12R_s^{-1}N^2p^2\varepsilon_{adder} + 4(N-1)p^2\varepsilon_{adder} + \alpha N^2\varepsilon_{adder}, \qquad (13)$$
$$\text{where } \alpha \in \{9,12,11\}.$$

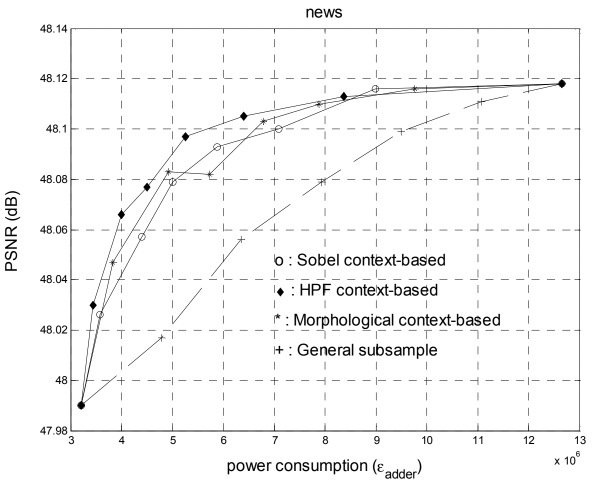


Fig. 9. The PSNR vs. power consumption of the News clip.

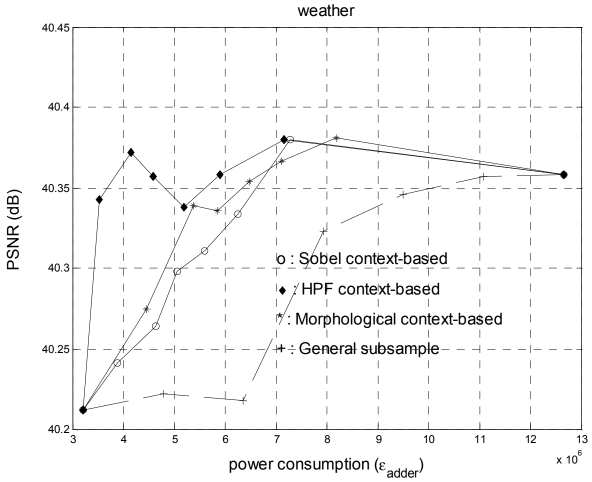


Fig. 10. The PSNR vs. power consumption of the Weather clip.

From (13), we can learn that the power overhead of EXU is a little. For the worst case, when the subsample rate is 4-to-1 and the gradient filter is Sobel-type, the power overhead is only 0.36% for the motion estimation with N=16 and p=32.

VI. Simulation Results

Figure 9 and 10 demonstrate the simulation results of two 352-by-288 MPEG clips for N=16 and p=32. The dashed lines are the results of the general subsample algorithm and the subsample rates at the bullets are (4:1), (8:3), (2:1), (8:5), (4:3), (8:7), and (1:1), respectively from left to right. The solid lines are the results of the content-based subsample algorithm with three gradient filters and the subsample rate is from (4:1) to (1:1). The threshold parameters, (m_1, m_2) pairs, at the bullets on solid lines are (1,0), (0.75,0.25), (0.5,0.5), (0.4,0.6), (0.3,0.7), (0.2,0.8), (0.1,0.9), and (0,1), from left to right respectively. As the results, the power consumption can be significantly reduced while the quality degradation is little and the power mode can be dynamically switched by simply changing the threshold parameters.

VII. Conclusion

We proposed the vario-power architecture based on a novel content-based subsample algorithm. The vario-power architecture provides the real-time capability for the switching of power consumption mode while the quality degrades a little. In the proposed architecture, the edge extraction unit plays a key role to dynamically adjust the power consumption mode and its power overhead can be neglected. As shown in the simulation results, the proposed algorithm successfully improves the compression quality of the general subsample algorithm and switches the power consumption mode by the content dependent technique.

References

[1] "Mobile Pentium III Processor in BGA2 and Micro-PGA2 Packages Datasheet," Intel Corporation, p55.

[2] K. M. Yang, M T. Sun, and L. Wu, "A family of VLSI designs for the motion compensation block-matching algorithm," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 36, no. 10, pp. 1317-1325, Oct. 1989.

[3] Yeong-Kang Lai, and Liang-Gee Chen, "A data-interlacing architecture with two-dimensional data-reuse for full-search block-matching algorithm," *IEEE Trans. Circuits syst. Video Technol.*, Vol. 8, no. 2, pp. 124-127, Apr. 1998.

[4] Chaur-Heh Hsieh, and Ting-Pang Lin, "VLSI Architecture for Block-Matching Motion Estimation Algorithm," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 2, no. 2, pp. 169-175, Jun 1992.

[5] Jen-Chieh Tuan, Tian-Sheuan Chang, and Chein-Wei Jen, " On the Data Reuse and Memory Bandwidth Analysis for Full-Search Block-Matching VLSI Architecture," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 12, no. 1, pp. 61-72, Jan. 2002.

[6] Mei-Juan Chen, Liang-Gee Chen, and Tzi-Dar Chiueh, "One-dimensional full search motion estimation algorithm for video coding," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 4, no. 5, pp. 504-509, Oct. 1994.

[7] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishigura, "Motion compensated interframe coding for video conferencing," in *Proc. NTC'81*, New Orleans, LA, pp. G5.3.1-G5.3.5, Nov. 1981.

[8] J. R. Jain, and A. K. Jain, "Displacement measurement and its application in interframe image coding," IEEE Trans. Commun. Vol. COM-29, pp. 1799-1808, Dec. 1981.

[9] Renxiang Li, Bing Zeng, and Ming L. Liou, "A new three-step search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 4, no. 4, pp. 438-442, Aug. 1994.

[10] Ken Sauer, and Brian Schwartz, "Efficient block motion estimation using integral projections," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 6, no. 5, pp. 513-518, Oct. 1996.

[11] Viet L. Do, and Kenneth Y. Yun, "A Low-Power VLSI Architecture for Full-Search Block-Matching Motion Estimation," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 8, no. 4, pp. 393-398, Aug. 1998.

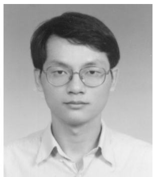
[12] W. Li, and E. Salari, "Succesive elimination algorithm for motion estimation," *IEEE Trans. Image Processing*, Vol. 4, no. 1, pp. 105-107, Jan. 1995.

[13] Wujian Zhang, Runde Zhou, and Kondo, T., "Low-power motion-estimation architecture based on a novel early-jump-out technique," *The IEEE International Symposium on Circuits and Systems*, Vol. 5 , pp. 187-190, 2001.

[14] Bede Liu, and A. Zaccarin, "New fast algorithms for the estimation of block motion vectors," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 3, no. 2, pp. 148-157, Arp. 1993.

[15] Chok-Kwan Cheung, and Lai-Man Po, "Normalized partial distortion search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 10, no. 3, pp. 417-422, Arp. 2000.

[16] Rafael C. Gonzalez, and Richard E. Woods, "Digital Image Processing," Addison Wesley, Sep. 1993.



Hsien-Wen Cheng received the B.S. degree in Control Engineering from National Chiao Tung University, Hsinchu, Taiwan, R.O.C. in 1992. He joined the AVerMedia Technologies Inc. from 1994 to 2001.

He is currently working toward the Ph.D degree in the Electrical and Control Engineer, National Chiao Tung University. His research interests are video/image compression, motion estimation, VLSI architecture and digital signal processing.

Lan-Rong Dung received a BSEE and the Best Student Award from Feng Chia University, Taiwan, in 1988, an MS in electronics engineering from National Chiao Tung University, Taiwan, in 1990, and Ph.D. in electrical and computer engineering from Georgia Institute of Technology, in 1997.

From 1997 to 1999 he was with Rockwell Science Center, Thousand Oaks, CA, as a Member of the Technical Staff. He joined the faculty of National Chiao Tung University, Taiwan in 1999 where he is currently an assistant professor in the Department of Electrical and Control Engineering. He received the VHDL International Outstanding Dissertation Award celebrating in Washington DC in October, 1997. His current research interests include VLSI design, digital signal processing, hardware-software codesign, and System-on-Chip architecture. He is a member of Computer and Signal Processing societies of the IEEE.

A POWER-AWARE ME ARCHITECTURE USING SUBSAMPLE ALGORITHM

Hsien-Wen Cheng and Lan-Rong Dung

The Department of Electrical and Control Engineering, National Chiao Tung University

ABSTRACT

This paper presents a power-aware architecture driven by a novel content-based subsample algorithm which allows the architecture to work at different power consumption modes with acceptable and smooth quality degradation. The proposed algorithm first performs the edge extraction to generate a turn-off mask which is used to reduce the switch activities of processing elements (PEs) in the semi-systolic array. Since we introduce an adaptive control mechanism to set the threshold value of edge determination, based on the video content and the remaining capacity of battery pack, the reduction of the switch activities is rather stationary at a certain power consumption mode. As shown in simulation results, the architecture can dynamically operate at different power consumption modes with little quality degradation while the power overhead of edge extraction is under 0.8% comparing with the general subsample algorithm.

1. INTRODUCTION

Motion estimation (ME) has been notably recognized as the most critical part in many video compression systems, such as MPEG standards and H.26x, which tends to dominate computational load and hence power requirements. With increasing demand of portable, battery-powered multimedia devices, a power-aware motion estimation architecture that can be flexible in both power consumption and compression quality is highly required. To meet the power-aware requirement of portable devices, this paper presents a power-aware ME architecture using a novel content-based subsample algorithm, that can adaptively perform trade-offs between power consumption and compression quality as the battery status changes [1]. Since the control mechanism and data sequences at different power consumption modes are the same, the power-aware architecture can perform the switching of power consumption mode very smoothly on the fly.

Lots of published papers presented efficient algorithms for the VLSI implementation of motion estimation but most proposed algorithms cannot dynamically adapt to different power consumption modes. Among the proposed algorithms, the Full-Search Block-Matching (FSBM) algorithm with Sum of Absolute Difference (SAD) criterion is the most popular approach for motion estimation because of its considerably good quality [1][2][3]. However, a huge number of comparison/difference operations result in high computation load and power consumption. To reduce the computational complexity of FSBM, researchers have proposed various fast algorithms that either reduce search steps [4] [5] [6] or simplify the calculation of error criterion [7]. The fast-search algorithms improve the block matching speed while the quality degradation is little and, thus, lead to a low power implementation. However, a low power implementation is not necessarily a power-aware system in that a power-aware system should adaptively modify its behavior with the change of power/energy status and balance the performance between quality and battery life [8].

Articles in [9] proposed a subsample algorithm to significantly reduce the computation cost with low quality degradation and the power consumption can be reduced by simply increasing the subsample rate; thus, the subsample algorithms are very suitable for power-aware ME architecture. To alleviate the aliasing problem of subsample, this paper presents a content-based algorithm which first uses edge extraction techniques to separate the high-frequency band from a macro-block and then performs subsampling within the low-frequency band. By an adaptive control mechanism, the edge-extraction step can self-tune the threshold value to maintain a stationary subsample rate. Based on the proposed algrorithm, we present a semi-systolic architecture which can dynamically alter the subsample rate as the power consumption mode changes. As the result, our methodology successfully switches the power mode while the quality degradation is little.

2. CONTENT-BASED SUBSAMPLE ALGORITHM

As mentioned above, the general subsample algorithm has aliasing problem for high subsample rate which leads to considerable quality degradation because the high frequency band is messed up. To alleviate the problem, this paper presents a content-based subsample algorithm whose pro-

This work was supported in part by Taiwan MOE Program for Promoting Academic Excellent of Universities under the grant number 91-E-FA06-4-4 and the National Science Council, R.O.C., under the grant number NSC 92-2220-E-009-003.

[1] The idea is motivated from the SpeedStep$^{\text{TM}}$ technology of Intel®.

```
//frame:$t$
Input current and reference frames, $W \times H$ ;
for($y = 0; y < W/N; y{+}{+}$){
  for($x = 0; x < H/N; x{+}{+}$){
    Perform gradient filtering;
    $threshold = m_1^t(x,y) \cdot \max\{G(i,j)\} + (1 - m_1^t(x,y)) \cdot \min\{G(i,j)\}$
    $EdgeMask(i,j) = \begin{cases} 1, G(i,j) \ge threshold \\ 0, \quad otherwise \end{cases}$
    $CSM(i,j) = SM_{4:1}(i,j) \vee EdgeMask(i,j)$;
    $csm\_cnt = total\ edges\ of\ CSM$;
    //update threshold parameter for the next frame
    $m_1^{t+1}(x,y) = m_1^t(x,y) + K_p \cdot (csm\_cnt - trg\_cnt)$;
    $if\ (m_1^{t+1}(x,y) < 0)\ \{m_1^{t+1}(x,y) = 0\}$;
    $if\ (m_1^{t+1}(x,y) > 1)\ \{m_1^{t+1}(x,y) = 1\}$;
    //find MV
    $SSAD_{\min}(x,y) = \infty$;
    for($u = -p; u < p; u{+}{+}$){
      for($v = -p; v < p; v{+}{+}$){
        $SSAD(u,v) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} \left| CSM(i,j) \cdot (S(i+u, j+v) - R(i,j)) \right|$;
        if $SSAD_{\min}(x,y) > SSAD(u,v)$
          {$SSAD_{\min}(x,y) = SSAD(u,v); MV(x,y) = (u,v)$; }
      }//for loop index $v$
    }//for loop index $u$
  }//for loop index $x$
}//for loop index $y$
```

Fig. 1. The content-based subsample algorithm

cedure is described in Fig. 1. The proposed algorithm first uses a edge extraction technique to separate edge pixels from a macro-block and performs subsampling to remaining pixels, then generates the content-based subsample mask (CSM). Upon the CSM generated, we are able to calculate SSAD values and find the best motion vector.

The determination of edge pixels starts from gradient filtering. This paper uses three popular gradient filters [10] to exercise the content-based algorithm: the high-pass filter, the Sobel filter, and the morphological gradient filter. After obtaining the gradients G from the filter, the algorithm determines the edge threshold of this CMB by an adaptive control mechanism. Then, it uses this threshold value to pick the edge pixels and produce the edge mask. Finally, the contend-based subsample mask (CSM) is generated and, therefore, the content-based subsample rate (CSR) denoted as R_s is N^2-to-csm_cnt, where csm_cnt means the number of 1's in CSM.

According to battery status, the host processor decides the desired subsample rate, that is N^2-to-trg_cnt, where trg_cnt is the target number of 1's in CSM). To make the subsample rate stationary, the algorithm adaptively updates the threshold-parameter $m_1^{t+1}(x,y)$ based on the difference of csm_cnt and trg_cnt and the control parameter K_p which will affect the settling time and steady-state error of subsample rate. Thus, the switching of power mode can be precisely done by giving the target subsample pixel count trg_cnt according to the battery status.

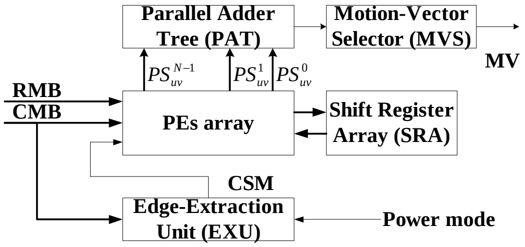

Parallel Adder Tree (PAT)
Motion-Vector Selector (MVS)
MV
PS_{uv}^{N-1}
PS_{uv}^{1}
PS_{uv}^{0}
RMB
CMB
PEs array
Shift Register Array (SRA)
CSM
Edge-Extraction Unit (EXU)
Power mode

Fig. 2. The system block diagram

3. THE POWER-AWARE ARCHITECTURE

According to the content-based subsample algorithm, we present a semi-systolic architecture shown in Fig. 2, based on architecture proposed by Hsieh and Lin [2], that contains an edge-extraction unit (EXU), an array of processing elements (PEs), a parallel adder tree (PAT), a shift register array (SRA), and a motion-vector selector (MVS). Given the power consumption mode, the EXU extracts high frequency (or edge) pixels from the current macro-block (CMB) and generates 0-1 content-based subsample masks (CSM) for the PE array to disable or enable processing elements (PEs). The PE array is used to accumulate absolute pixel differences column by column while the parallel adder tree sum up all the results to generate the value of subsample sum of absolute difference (SSAD). The MVS, then, performs compare-and-select operation to select the best motion vector.

Based on the semi-systolic architecture with content-based subsample algorithm, the architecture dynamically disable some processing elements to reduce the power consumption in that we assume the major power consumption is determined by the switch activity of system. By performing the edge extraction first, a target count (trg_cnt) was then set as the criterion of whether to enable/disable processing elements and thus dynamically changes the switch activities of system to reduce the power consumption. Fig. 3 shows the PE structure and explains how the CSM disables/enables processing elements. The CSM disables the PE by using the block element (BE), implemented by AND gates, that can nullify the input signals of data path, that consists of the absolute difference unit ($|a - b|$) and the Adder unit. That is, the circuits in these units remain still until the next motion vector searching iteration starts and, thus, the consumption

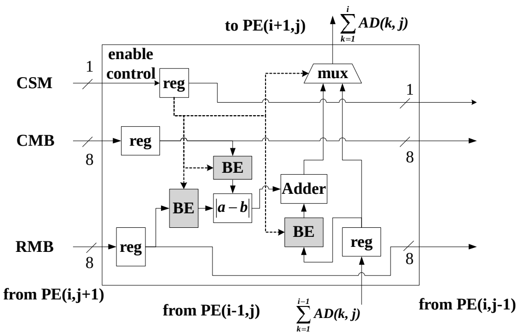


Fig. 3. The structure of a PE.

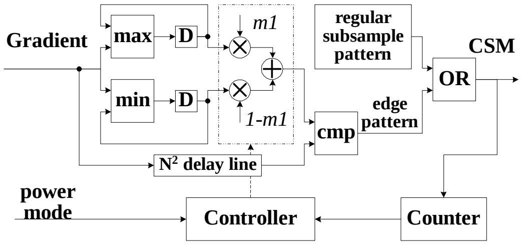


Fig. 4. The architecture of edge-determination and CSM generator.

of transient power can be saved. The edge-extraction unit contains two blocks, that are gradient filter and CSM generator. Fig. 4 illustrates the structure of CSM generator.

4. RESULTS

Figures 5 and 6 demonstrate the simulation results of two 352-by-288 MPEG clips for $N = 16$ and $p = 32$ and the control parameter K_p is set as 0.3. The target subsample pixel counts are set as 64, 96, 128, 160, 192, 224 and 256, respectively. The dashed lines are the results of the general subsample algorithm and the solid lines are the results of the content-based subsample algorithm with three gradient filters. As shown in the results, the quality degradation of the content-based algorithm is less than that of the general subsample algorithm, and the type of selected gradient filter does not make much difference to the performance of the proposed algorithm. From the results, the average CSR error is as low as 1.12% and the CSR error variance is as low as 0.00024. Thus, the proposed algorithm can be applied for power-aware system in that the subsample rate can be nearly stationary with given target subsample rate.

One can consider the major power consumption of a CMOS gate $gate_i$ as (1), where α and κ are constants, C_i is the output capacitance of gate i, f_i is the operation frequency of gate i, and $r_i(0 \leftrightarrow 1)$ is the switch activity of gate i. For an execution unit EU_j in a VLSI system, the power consumption can be shown in (2), where $N_{gate,j}$ is the gate count of EU_j.

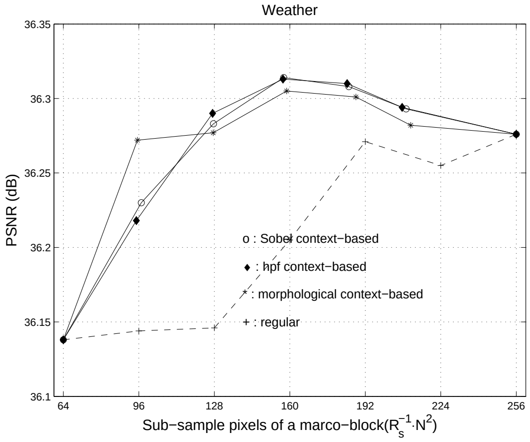


Fig. 5. The quality degradation of the weather clip.

$$P_{gate_i} = \alpha \cdot C_i \cdot f_i \cdot V_{DD}^2 = \kappa \cdot C_i \cdot r_i(\updownarrow). \quad (1)$$

$$P_{EU_j} = \sum_{i=1}^{N_{gate,j}} \kappa \cdot C_i^j \cdot r_i^j(\updownarrow). \quad (2)$$

After considering the activity of execution units, the total power consumption can be as (7) by assuming the switch activities are uniform within an execution unit; that is, $r_i^k(\updownarrow) = r^k(\updownarrow), \forall r_i^k(\updownarrow)$ and the average output capacitances are nearly same. In this paper, we use the gate power coefficient ε_{gp} as the unit for estimating power dissipation.

$$\begin{aligned} P_{total} = & \sum_{\forall \text{inactive} EU_j} P_{EU_j} + \sum_{\forall \text{active} EU_k} P_{EU_k} && (3) \\ \cong & \ \kappa \sum_{\forall \text{active} EU_k} r^k(\updownarrow) \sum_{i=1}^{N_{gate,k}} C_i^k && (4) \\ = & \ \kappa \cdot \sum_{\forall \text{active} EU_k} r^k(\updownarrow) \times C_{avg}^k \times N_{gate,k} && (5) \\ \cong & \ (\kappa \cdot C_{avg}) \sum_{\forall \text{active} EU_k} r^k(\updownarrow) \times N_{gate,k} && (6) \\ = & \ \varepsilon_{gp} \sum_{\forall \text{active} EU_k} r^k(\updownarrow) \times N_{gate,k} && (7) \end{aligned}$$

Table 1 shows the synthesis result with the TSMC 1P4M $0.35um$ cell library, where the symbol R_s means the content-based subsample rate and the ε_{gp} is the gate power coefficient defined in (7). From the results, we can learn that the area overhead to implement EXU is 7.68% and the power overhead is only 0.8% in the worst case when the subsample rate is 4-to-1 for the motion estimation with $N = 16$ and $p = 32$.

Table 1. Power analysis of the power-aware architecture

EU_i	PE array		SRA	PAT+MVS	EXU
	AD + Adder	Others			
Gate Count G^i	117,760	58,708	44,640	1,800	17,121
$r^i(\updownarrow)$	$4p^2R_s^{-1} = 4096R_s^{-1}$	$4p^2 = 4096$	$4p^2 = 4096$	$4p^2 = 4096$	$N^2 = 256$
$P^i_{consumption}$	$4.8e8 \cdot R_s^{-1}$	$2.4e8$	$1.8e8$	$7.37e6$	$4.38e6$
$P^{all}_{consumption}(\varepsilon_{gp})$	$4.8e8 \cdot R_s^{-1} + 4.3e8$				

$N = 16$ and $p = 32$
Cell library: TSMC $0.35um$ process

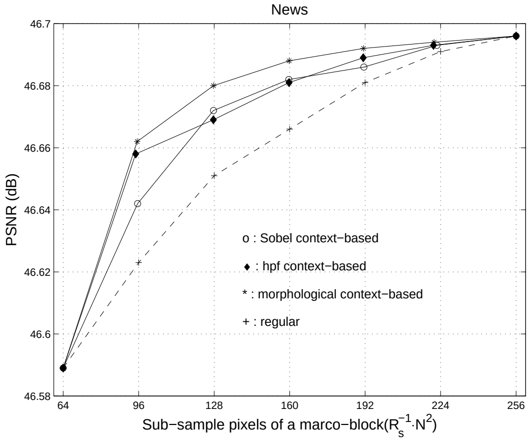


Fig. 6. The quality degradation of the news clip.

5. CONCLUSION

Motivated by the concept of battery properties and SpeedStep technology, this paper presents an architecture-level power-aware technique based on a novel content-based subsample algorithms. The switching of power consumption mode can be smooth and, thus, the proposed architecture can provide real-time capability for switching of power consumption mode with little quality degradation. As shown in the simulation results, the proposed algorithm successfully improves the compression quality of the general subsample algorithm and switches the power consumption mode by adaptively adjusting the threshold parameters.

6. REFERENCES

[1] K. M. Yang, M T. Sun, and L. Wu, “A family of VLSI designs for the motion compensation block-matching algorithm,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 36, no. 10, pp. 1317–1325, Oct. 1989.

[2] Chaur-Heh Hsieh and Ting-Pang Lin, “VLSI architecture for block-matching motion estimation algorithm,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 2, pp. 169–175, Jun. 1992.

[3] Jen-Chieh Tuan, Tian-Sheuan Chang, and Chein-Wei Jen, “On the data reuse and memory bandwidth analysis for full-search block-matching VLSI architecture,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 1, pp. 61–72, Jan. 2002.

[4] Mei-Juan Chen, Liang-Gee Chen, and Tzi-Dar Chiueh, “One-dimensional full search motion estimation algorithm for video coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, no. 5, pp. 504–509, Oct. 1994.

[5] J. Y. Tham, S. Ranganath, M. Ranganath, and A. A. Kassim, “A novel unrestricted center-biased diamond search algorithm for block motion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 4, pp. 369–377, Aug. 1998.

[6] Ce Zhu, Xiao Lin, and Lap-Pui Chau, “Hexagon-based search pattern for fast block motion estimation,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 5, pp. 349 –355, May 2002.

[7] Jeng-Hung Luo, Chung-Neng Wang, and Tihao Chiang, “A novel all-binary motion estimation (ABME) with optimized hardware architectures,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 8, pp. 700 –712, Aug. 2002.

[8] Osman S. Unsal and Israel Koren, “System-level power-aware design techniques in real-time systems,” *Proceedings of the IEEE*, vol. 91, no. 7, pp. 1055 –1069, July 2003.

[9] Bede Liu and A. Zaccarin, “New fast algorithms for the estimation of block motion vectors,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 2, pp. 148–157, Apr. 1993.

[10] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Addison Wesley, Sep. 1993.

PAPER

ON MULTIPLE-VOLTAGE HIGH-LEVEL SYNTHESIS USING ALGORITHMIC TRANSFORMATIONS

Lan-Rong DUNG[†] ***and*** **Hsueh-Chih YANG**[†], ***Nonmembers***

SUMMARY This paper presents a multiple-voltage high-level synthesis approach for low power DSP applications using algorithmic transformation techniques. Our approach is motivated by maximization of task mobilities in that the increase of mobilities may raise the possibility of assigning tasks to low-voltage components. The mobility means the ability to schedule the starting time of a task. It is defined as the distance between its as-late-as-possible (ALAP) schedule time and its as-soon-as-possible (ASAP) schedule time. To earn task mobilities, we use loop shrinking, retiming and unfolding techniques. The loop shrinking can first reduce the iteration period bound (IPB) and, then, the others are employed for shortening the iteration period (IP) as much as possible. The minimization of IP results in high task mobilities. Finally, we can assign tasks with high mobilities to low-voltage components and, thus, minimize energy under resource and latency constraints. With considering the overhead of level conversion, our approach can achieve significant power reduction. In the case of the third-order IIR filter, the proposed approach can save up to 40.2% of power consumption.

key words: *multiple voltage scheduling, low-power circuit, loop shrinking, retiming, unfolding, high-level synthesis*

1. Introduction

With the increasing demand of portable devices, the reduction of power consumption has become the essential issue in VLSI design. A growing literature on VLSI design has proposed to reduce power consumption at different levels, from physical level to system level. Since any design decision made at earlier stages will have higher impacts on the final result, researchers believe that the power minimization should be done at higher abstraction levels for more significant power saving [1], [2]. A number of papers have addressed on power saving techniques, such as voltage scaling, capacitance reduction and switching minimization, for high level synthesis (HLS) [2]–[5]. However, these papers are based on a single voltage supply for power minimization and cannot take full advantage of available schedule slack to reduce the voltage. Therefore, the use of multiple supply voltages becomes very attractive to low power design recently [6]–[13]. The idea is to assign non-critical nodes to low voltage components and execute time-critical nodes at higher supply voltage. Papers [2], [8], [11], [12] present multiple-voltage scheduling for power optimization for HLS. Using integer linear programming (ILP) or dynamic programming, their works have pseudo-polynomial or even exponential time complexity. In paper [10], Shiue and Chakrabarti present a list-based multiple-voltage scheduling algorithm with polynomial time complexity. The algorithm is driven by three parameters: depth, mobility, and switching capacitance. With considering the level shifter, [10] provides effective resource-constrained and latency-constrained schemes for multiple-voltage HLS. From Chakrabarti's group, later on, paper [13] uses Lagrange multiplier method to find the optimal solution of multiple-voltage scheduling under both resource and latency constraints.

The papers mentioned above have presented efficient scheduling for multiple-voltage HLS. Yet, few papers have considered the effect of algorithmic transformation on multiple-voltage power minimization. In this paper, we exploit on algorithmic transformation for multiple-voltage HLS and present an efficient approach to minimize power consumption under resource and latency constraints. The main concept behind the proposed approach is to change the computational structures by transformations and make mobility of each task in fully-specified flow graph (FSFG) as high as possible. The mobility means the ability to schedule the starting time of a task. It is defined as the distance between its as-late-as-possible (ALAP) schedule time and its as-soon-as-possible (ASAP) schedule time. Obviously, the increase of mobilities may raise the possibility of assigning tasks to low-voltage components. To earn

[†] The authors are with the Department of Electrical and Control Engineering, National Chiao-Tung University, Hsinchu 30010, Taiwan

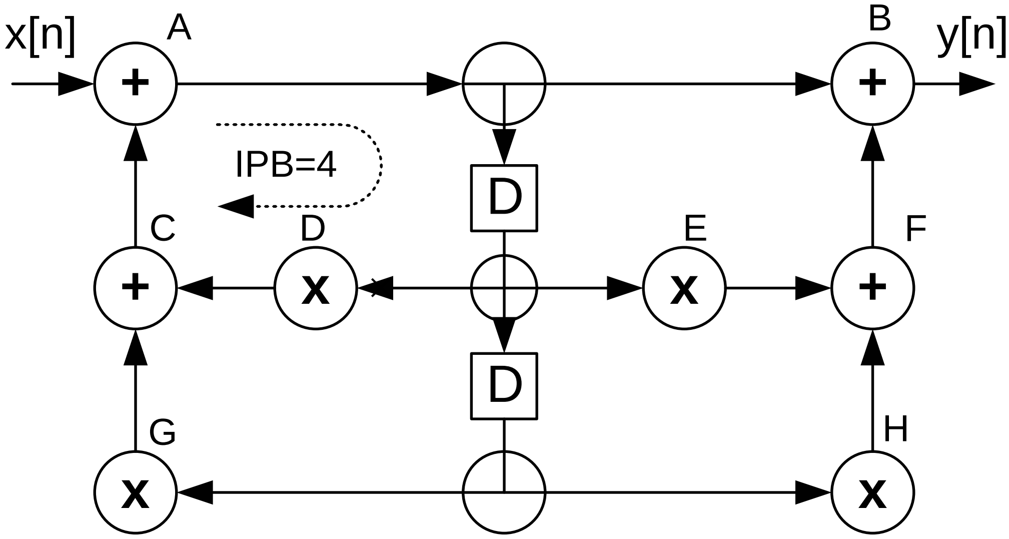


Fig. 1 FSFG of second-order IIR.

task mobilities, we use loop shrinking, retiming and unfolding techniques. First, we use algebraic transformation to shrink the loops and hence decrease the iteration period bound (IPB). Then, retiming and unfolding techniques are employed for shortening the iteration period (IP) as much as possible. The minimization of IP results in high task mobilities. Once the IP is minimized, we can then assign tasks with high mobilities to low-voltage components based on a proposed task-assignment scheme. The proposed scheme is priority-based in that tasks with higher effectiveness will be given higher scheduling priority. The effectiveness of a task is defined as the difference of energy consumption between its high-voltage assignment and low-voltage assignment. So, the most significant power reductions can be counted first. Finally, our approach uses a level-conversion refinement step to reduce the power overhead of using level converters as much as possible. As the results, our approach can achieve significant power reduction under resource and latency constraints. In the case of the third-order IIR filter, the proposed methodology can save up to 40.2% of power consumption while the supply voltages are 5V and 3.3V under latency constraint $1.5\dot{T}_{critical}$ and resource constraints 1, 1, 1, 1.

The rest of the paper is organized as follows. In Section 2, we introduce algorithmic transformations. Section 3 presents the proposed approach in details. Section 4 shows the experimental results and Section 5 is the conclusion of this work.

2. Algorithmic Transformations

2.1 Fully-Specified Flow Graph

This paper uses Fully-Specified Flow Graph (FSFG) as a design entry for describing algorithm. A deterministic DSP algorithm can be represented by a FSFG which describes the relationship between a set of input and output sequences [14]. The FSFG is defined by the 3-tuple $< N, E, D >$, where N is the set of vertices or nodes that represent operations on a process element (PE), E is the set of directed edges that describe the data flow and D is the set of ideal delays. Fig. 1 shows an FSFG for a second-order IIR filter. Given enough function units, the performance of FSFG is determined by loops [15].

In literature on HLS, the iteration period bound(IPB) has been used to measure the efficiency of the implementation of an FSFG [15]–[17]. The iteration period (IP) for a loop is defined as the total computational latency in the loop divided by the total number of delays. The IPB is the maximum IP for all loops in FSFG. The IPB in some sense represents the minimal sample period cycle at which a circuit can operate. In other words, the IPB represents the minimum achievable latency between iterations of the given flow graph when enough processors are

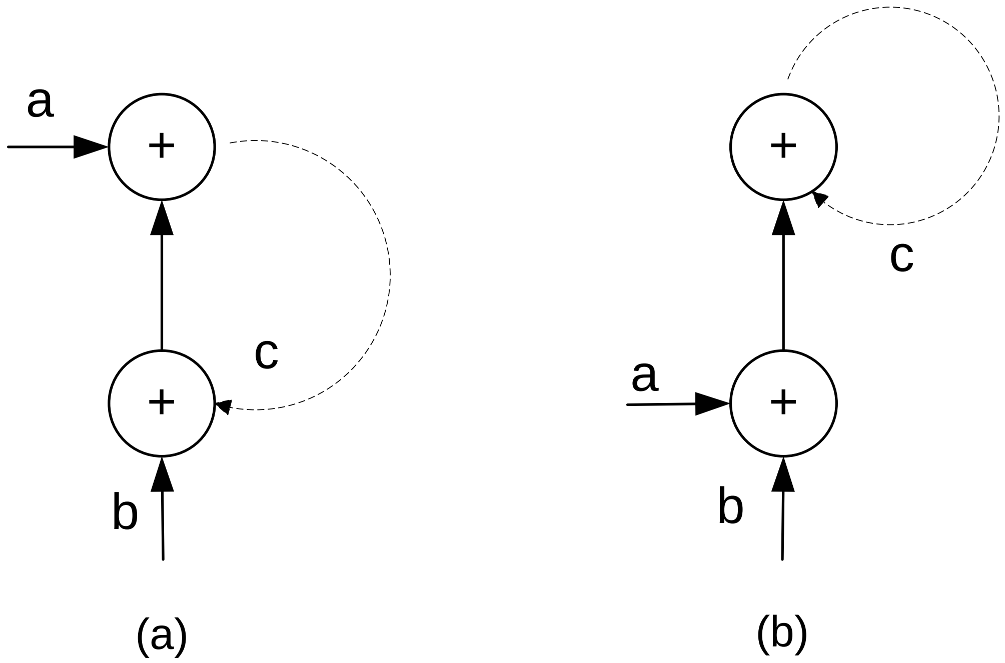


Fig. 2 Loop shrinking using associativity of addition. (a) The original FSFG. (b) The equivalent FSFG.

available. For instance, if we assume that a multiplication takes two time units to execute and an addition one time unit, the FSFG shown in Fig. 1 has an IPB of 4 time units.

A smaller IPB represents a higher throughput or sample rate if it can be achieved; however, the optimal sample period is not always achievable. Consider Fig. 1 again. The sample period is limited by loop $G - C - A - B$ and so equal to 5 time units. To obtain the rate-optimal implementation of FSFG, this paper uses algorithmic transformations to change computing structures and gain better performance in the stage of implementation. The following subsections will introduce three algorithmic transformation techniques employed in this paper.

2.2 Loop Shrinking

Loop shrinking can reconstruct the FSFG to obtain the optimal IPB for loops. Consider the loop segement shown in Fig. 2. Fig. 2 (a) has a chain of two additions within the loop. According to the associativity of addition, the function $a + (b + c)$ in Fig. 2 (a) is equivalent to the function $(a + b) + c$ in Fig. 2 (b). Obviously, the dot-lined loop has been shrunk in (b) and the loop latency is reduced as well. Therefore, we can perform loop shrinking on critical loop, who has the maximum IP, to reduce the IPB while the functionality of FSFG keeps the same. Fig. 3 is an example of loop shrinking. Given each task takes one time unit to execute, the IPB can be reduced from 3 time units to 2 time unites.

2.3 Retiming and Unfolding

Optimization of IPB may lead to a rate-optimal FSFG, but the optimal sample period is not guaranteed in an IPB-optimal FSFG. Retiming is a process that may help making sample rate equal to IPB. With delay transfer or nodal transfer, it is possible to make a loop achieve IPB. Integer linear programming (ILP) for retiming listed in Fig. 4 has been proposed to achieve the IPB [18]. The ILP formulation is attractive because additional constraints can easily be added to the formulation. Unfortunately, the ILP for retiming might take more computational time and cannot guarantee the achievement of IPB. Fig. 5, for example, the IPB can not be achieved by retiming since

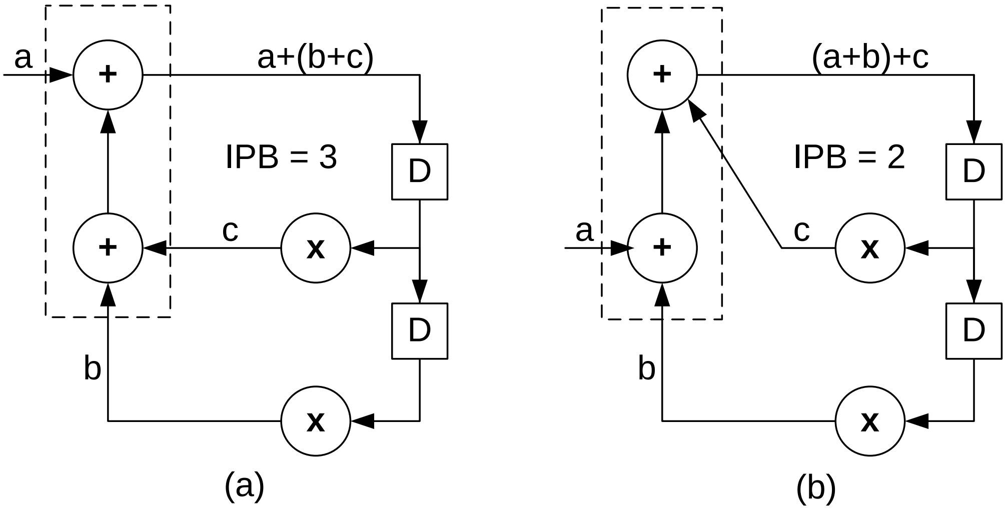


Fig. 3 Loop shrinking of Second-order IIR. (a) The original FSFG. (b) The equivalent FSFG.

Given FSFG,
Maximizing ($t_{\text{zero-delay-path}}$)
subject to:

$$r(u)\text{-}r(v) \leq D_i(e),$$

where e is the directed edge from v to u.

Fig. 4 ILP of retiming

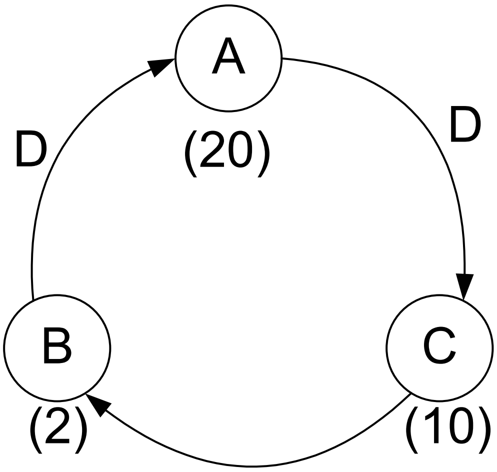


Fig. 5 An example of FSFG that cannot achieve IPB.

node A requires 20 time-unit to execute.

To guarantee the achievement of IPB and optimize the sample period, paper [17] presents unfolding technique. Instead of describing one iteration of the computation in the form of a recursive loop, unfolding by a factor P implies P consecutive iterations. Fig. 6 illustrates the unfolding result of FSFG in Fig. 5. In the unfolding FSFG, the IPB can be achieved and the sample rate is optimized.

3. Proposed Approach

3.1 Multi-Voltage HLS Algorithm

Fig. 7 describes the proposed HLS algorithm. The inputs to our algorithm are an FSFG, a resource constraint (Ru), and a latency constraint (Tu), and the outputs are the voltage assignment, start time, and end time of each node and the total power consumption of the scheduling if the scheduling exists. In a nutshell, the proposed resource and latency constrained algorithm operates in four passes. In the first pass, the input file specifies the resource constraint (Ru), the latency constraint (Tu), and the operations within the FSFG. Once having the input information, we use loop shrinking technique to reduce the IPB. This bound ensures that there exists a period that is sufficiently long to assure proper evaluation of all the loops. The consecutive iteration of the execution of the FSFG cannot begin before the IPB. In the second pass, we compute the IP to check whether it matches the IPB or not. If the IP matches the IPB, the graph will be sent to the pass three. If the IP is not equal to the IPB yet, the IP minimization can be achieved by either retiming or unfolding techniques to obtain optimal mobility under the given resource constraint. In the third pass, multiple-voltage scheduling subroutine, $MultV\_Schedule(graph, Ru, Tu, L)$, is used to schedule and assign tasks to the proper scheduling time and the available resources such that the total

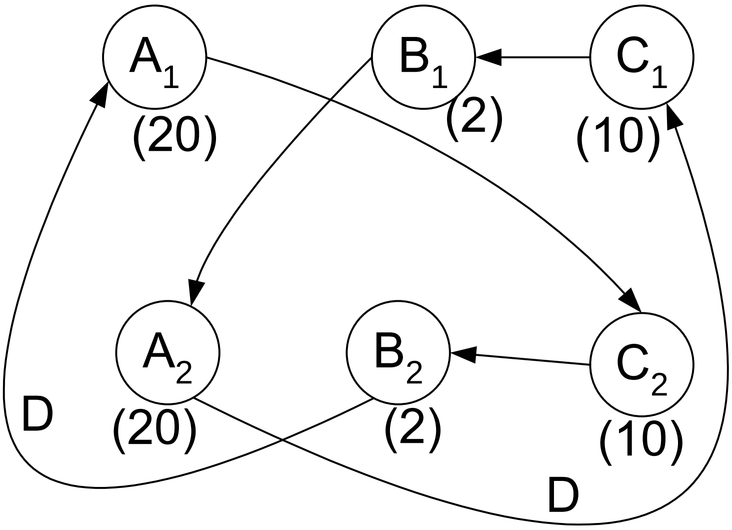


Fig. 6 A rate-optimal FSFG using unfolding.

```
SCHEDULE(FSFG, Ru, Tu, L, EnergyTb, LCTb)
// Ru: Resource constraint, Tu: Latency constraint
// L : number of voltage levels
// EnergyTb: the energy table of multiple voltages
// LCTb: the energy table of level converters
{
        g = Read_FSFG (FSFG, Ru, Tu);
        g1 = Shrink (g);
        if ( IP = IPB )   // Check if IP equals to IPB in g1
              S = MultV_Schedule (g1, Ru, Tu, L);
       else{
             g2 = Minimize_IP (g1);
             S = MultV_Schedule (g2, Ru, Tu, L);
          }
S = LC_refine (S); // refinement of level converters
Report ( S );
```

Fig. 7 The multi-voltage HLS algorithm.

power/energy consumption is minimum. In the last pass, $LC\_refine(S)$ considers the overhead power consumption caused by level converters by a heuristic methodology.

3.2 Loop Shrinking

We propose the loop shrinking flow as shown in the Fig. 8. The flow will search two adjacent addition operations in the critical loop first and then rearrange the associated edges to reduce the number of nodes in the critical loop. The flow will repeat calculating and updating the IPB of the critical loop until IPB cannot be improved. Once the optimal IPB can be obtained, we have higher chance to obtain higher mobility of each node in the scheduling to save more power consumption.

3.3 IP Minimization

IP minimization $Minimize\_IP(graph)$ can be achieved by either retiming or unfolding techniques to obtain optimum mobility under given resource constraints. We apply the unfolding technique to guarantee that IP matches the IPB, and minimize sample period or latency. Figures 9- 11 illustrate the ASAP and ALAP scheduling results

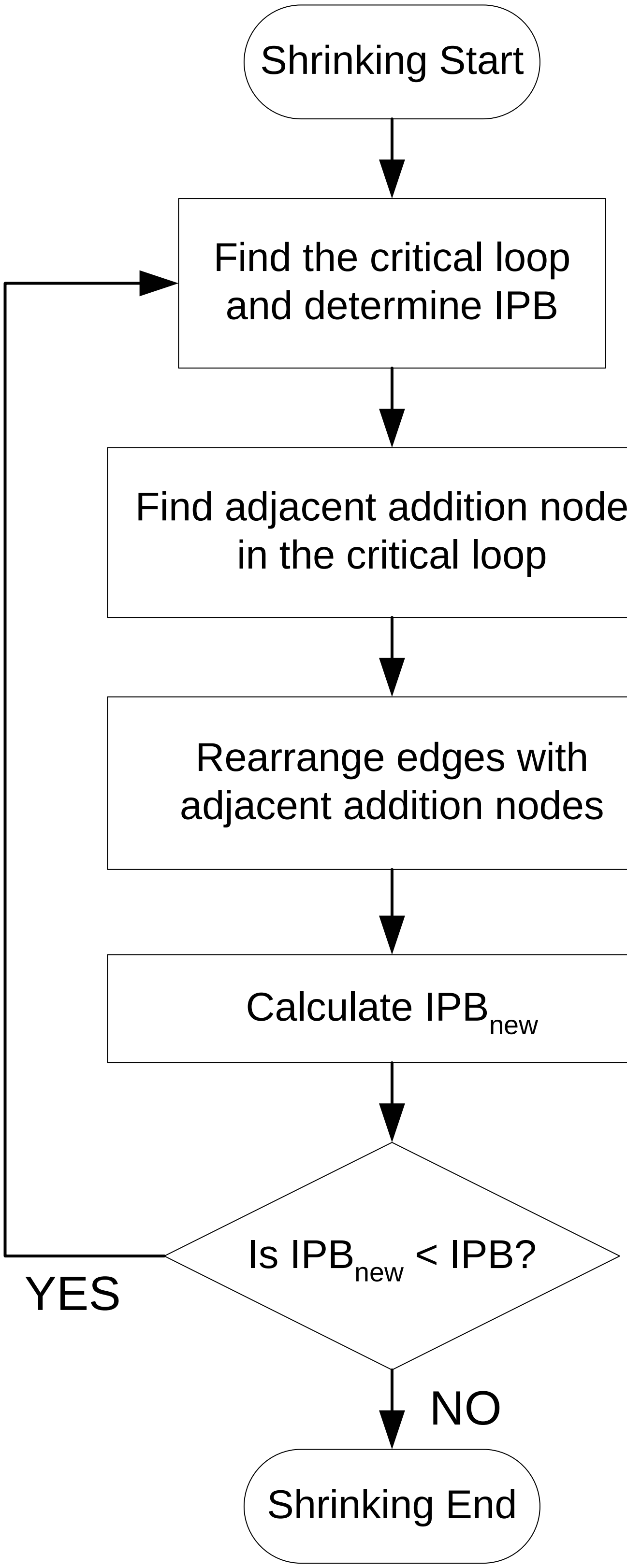

Shrinking Start
Find the critical loop and determine IPB
Find adjacent addition nodes in the critical loop
Rearrange edges with adjacent addition nodes
Calculate IPB$_{new}$
Is IPB$_{new}$ < IPB?
YES
NO
Shrinking End

Fig. 8 The loop-shrinking flow.

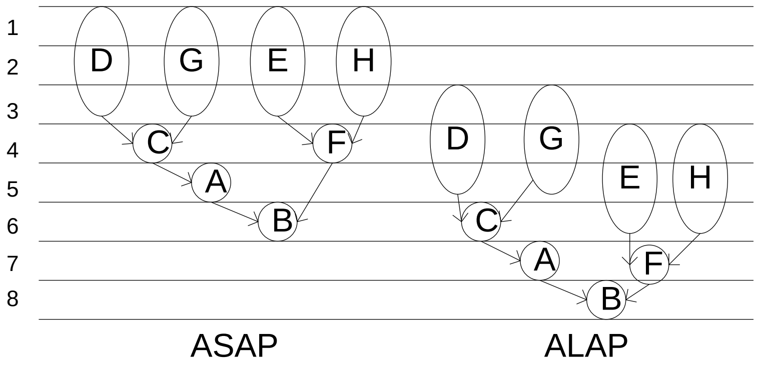


Fig. 9 Schedules of second-order IIR before retiming.

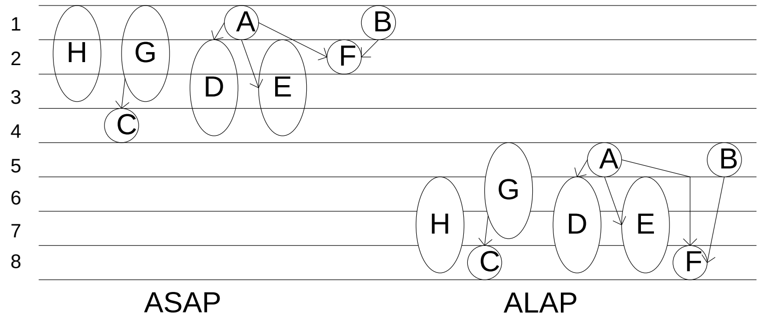


Fig. 10 Schedules of second-order IIR after retiming.

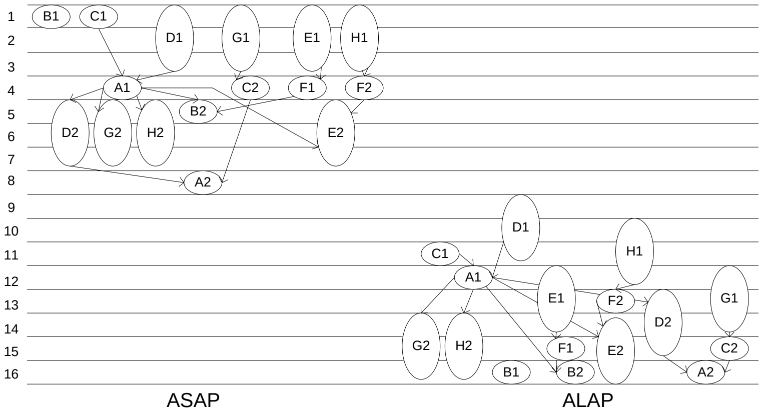
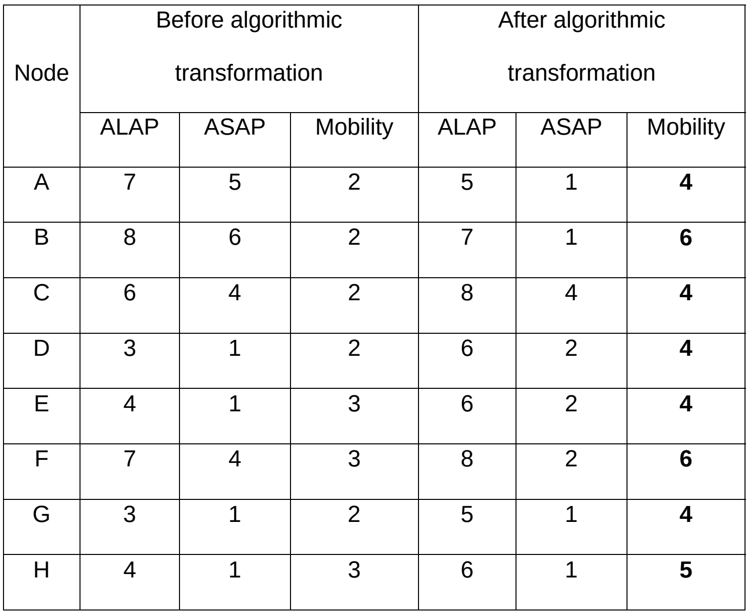


Fig. 11 Schedules of second-order IIR after unfolding.

of second order IIR filter using retiming and unfolding techniques. The tables in Fig. 12 and Fig. 13 shows that the mobilities are improved by algorithmic transformation.

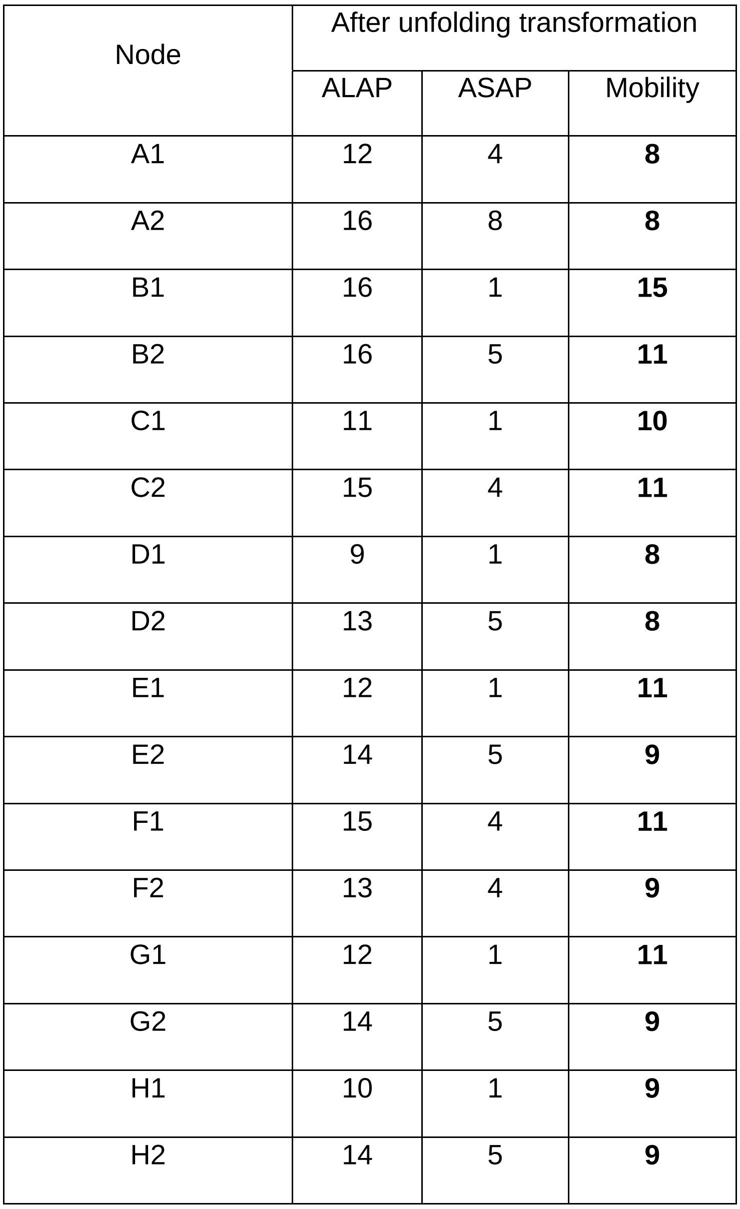
3.4 Multiple Voltage Scheduling

In this section, we explain the multiple-voltage scheduling subroutine, $MultV_Schedule(graph, Ru, Tu, L)$, where Ru represents the resource constraint, L represents the number of voltage levels, and Tu represents that the number of available scheduling slots equals to the latency constraint, or else equals to P times the latency constraint while the unfolding was used to make sure the IP and the IPB are the same in the second pass. Fig. 14 shows the flowchart of $MultV_Schedule(graph, Ru, Tu, L)$ where the index i, and j represents different classes of function units and voltage



Node	Before algorithmic transformation			After algorithmic transformation		
	ALAP	ASAP	Mobility	ALAP	ASAP	Mobility
A	7	5	2	5	1	**4**
B	8	6	2	7	1	**6**
C	6	4	2	8	4	**4**
D	3	1	2	6	2	**4**
E	4	1	3	6	2	**4**
F	7	4	3	8	2	**6**
G	3	1	2	5	1	**4**
H	4	1	3	6	1	**5**

Fig. 12 Mobilities of second-order IIR after retiming.



Node	After unfolding transformation		
	ALAP	ASAP	Mobility
A1	12	4	**8**
A2	16	8	**8**
B1	16	1	**15**
B2	16	5	**11**
C1	11	1	**10**
C2	15	4	**11**
D1	9	1	**8**
D2	13	5	**8**
E1	12	1	**11**
E2	14	5	**9**
F1	15	4	**11**
F2	13	4	**9**
G1	12	1	**11**
G2	14	5	**9**
H1	10	1	**9**
H2	14	5	**9**

Fig. 13 Mobilities of second-order IIR after unfolding.

levels, respectively. The index k represents the number of nodes in the class c. In the beginning, all nodes in the flow are set to be "unmarked" to represent the un-scheduled status of each node. Then the program will choose a class of operations, such as multiplications, according to the effectiveness priority of each class. The effectiveness of a task is defined as the difference of energy consumption between its high-voltage assignment and low-voltage assignment. This scheme is priority-based in that tasks with higher effectiveness will have higher priority. So, the assignment with most significant power reductions can be considered first. The next step is to compute the as-soon-as-possible (ASAP) value, the as-late-as-possible (ALAP) value, depth, and mobility for each node. Here the ASAP and the ALAP times are computed using the user defined latency constraint. Recursively assign nodes with higher priority to lower voltage resources and update the ASAP, ALAP value. Notice that the scheduling order of nodes does not always obey the data precedence order, which means we might deal with all the multiplication nodes before all addition nodes, therefore, we have to check the latency illegality in the scheduling process. Once the latency constraint was broken, the higher voltage resource will be utilized to make sure the scheduling result is feasible.

For example, there exists four multiplications and four additions for a second order IIR filter. Then we determine the "magic number", N, which represents the number of tasks in the function unit class with the highest effectiveness which we try to assign to the lowest voltage to save power dissipation as much as possible under the given constraint. The magic number N is determined by $\lfloor \frac{T_u}{T_c(v(j))} \rfloor$, where T_u is the latency constraint and $T_c(v(j))$ represents the execution time of the operation type with the highest effectiveness operating at $v(j)$ voltage. For instance, according to the Fig. 15, the multiplication nodes have higher effectiveness than that of addition nodes. If the latency constraint $Tu = 10 timeunits$ and the resource constraints are (1, 1, 1, 1) (one 3.3V multiplier, one 5V multiplier, one 3.3V adder, and one 5V adder), there are two ($\lfloor \frac{10}{4} \rfloor$) multiplications using 3.3V multipliers to save the power consumption because one 3.3V multiplier takes 4 time-unit delay and maximally assigning the nodes with the highest effectiveness within the latency constraints is the first step to save power consumption. If there is insufficient number of 3.3V multipliers, we compute the priority and higher priority nodes use the 3.3V multiplier first. The priority of a node is a function of its depth, mobility, ASAP, and ALAP. Depth is the most important parameter, since it is directly linked to the latency. We assign nodes with larger depth first, so the scheduling can be completed within the latency constraint. Mobility is the second most important parameter. Nodes with higher mobility are given higher priority during assignment of lower voltage resources. Nodes with smaller ASAP time can be scheduled earlier. Since the power consumption of the multiplier is much more than of the adder, we give the higher priority to the nodes with smaller ASAP time and arrange the scheduling first. Since we schedule multiplications before additions, it is essential to check if the scheduling legality by ALAP time and the function unit utilization obey the constraints. For example, in 3.3V operation, we need to guarantee that the start time must be smaller or equal to the original ALAP time-1. The reason is that 3.3V resources take one more delay than 5V resources do. This checking process depends on the delay of different function units. Note that the number of the multiplications which we try to assign to lowest voltage multipliers based on the given latency constraint and the resource constraint can not be achieved. Thus we must relieve the constraint by resetting the number of the multiplications assigned to lowest voltage multipliers and the number of the multiplications assigned to higher voltage multipliers.

3.5 Refinement of Level Converters

The proposed scheduling algorithm is based on multiple-voltage methodology, and, thus, the power consumption of level converters can not be ignored. According to Fig. 15 and Fig. 16, we realize that the power consumption of a 3.3V-to-5V level converter is much greater than what a 3.3V adder takes. Note the left-hand side of Fig.17, one task assigned to low voltage was scheduled between two tasks assigned to high voltage and the other task assigned to high voltage was scheduled between two tasks assigned to low voltage. There are four level converters will be required in the situation. We use the refinement method shown in Fig.18 to search this scheduling part and avoid the overhead power consumption caused by level converters as much as possible.

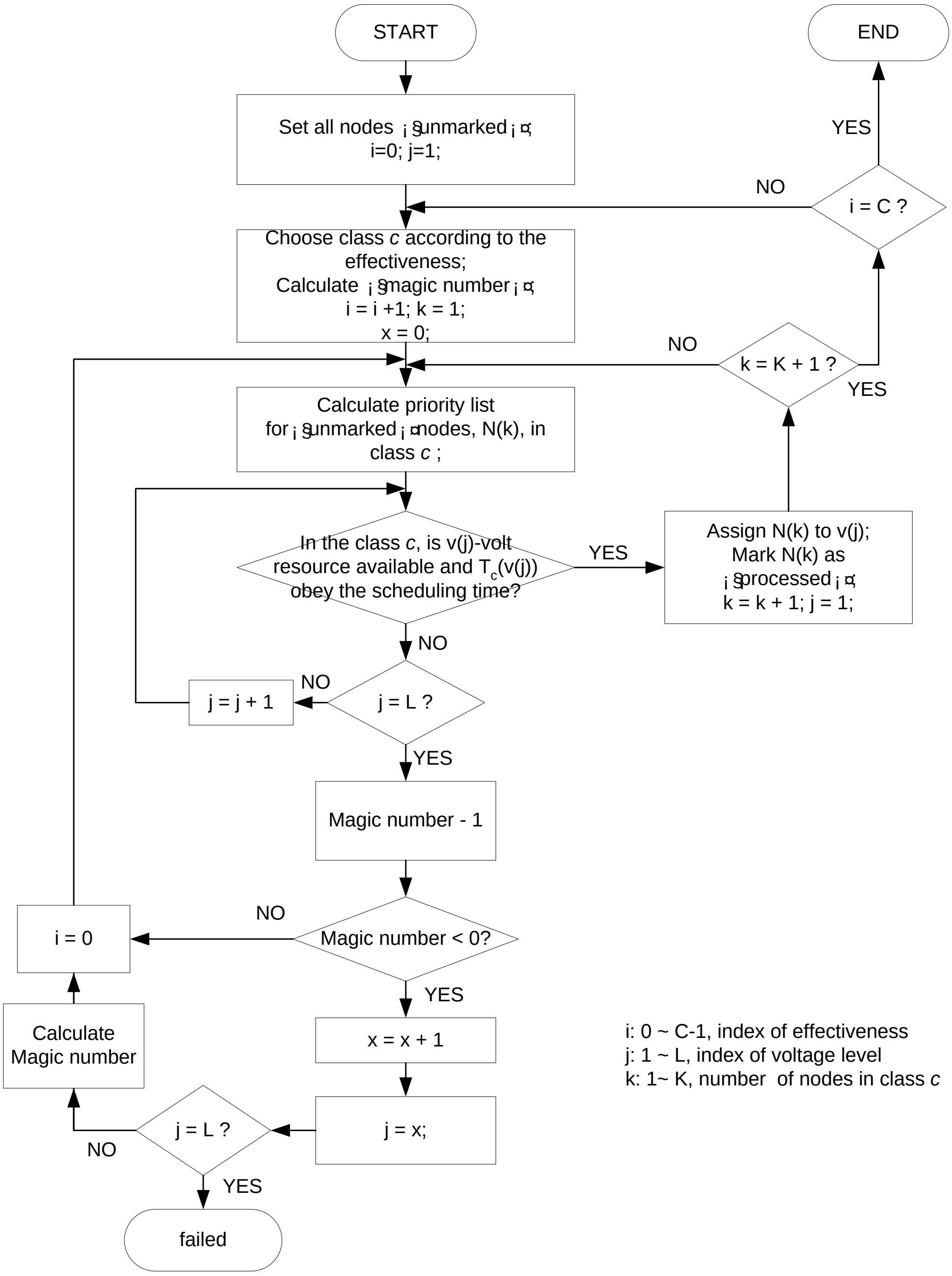
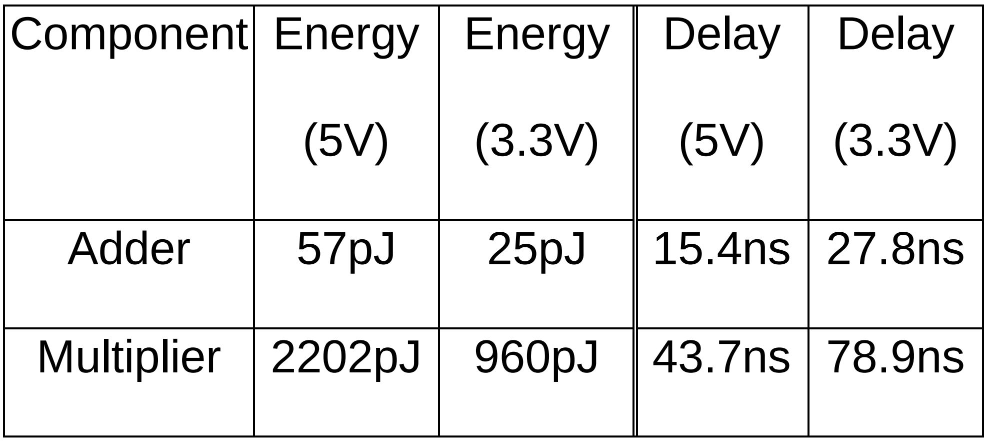
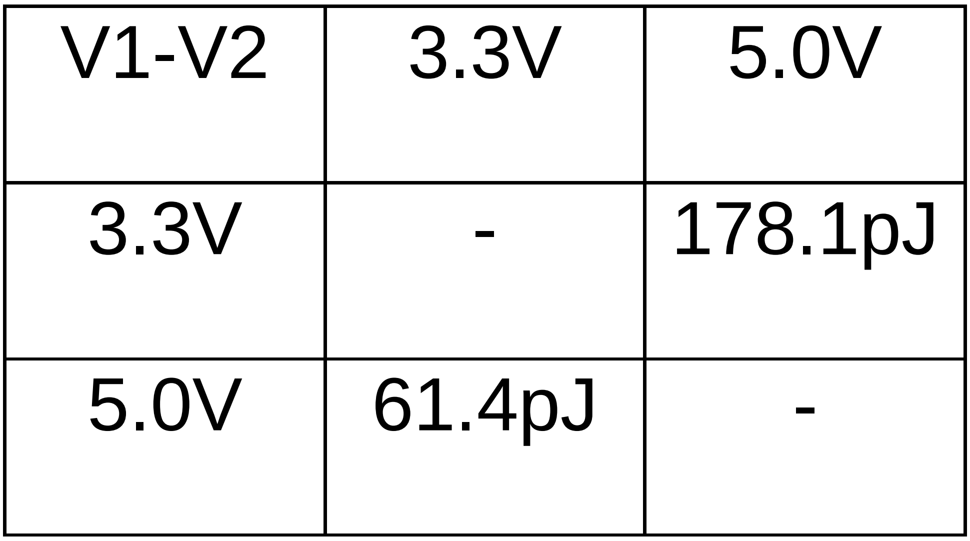

START
END
Set all nodes ¡§unmarked¡¨
i=0; j=1;
YES
NO
i = C ?
Choose class *c* according to the effectiveness;
Calculate ¡§magic number¡¨
i = i +1; k = 1;
x = 0;
NO
k = K + 1 ?
YES
Calculate priority list for¡§unmarked¡¨nodes, N(k), in class *c* ;
In the class *c*, is v(j)-volt resource available and T_c(v(j)) obey the scheduling time?
YES
Assign N(k) to v(j);
Mark N(k) as ¡§processed¡¨
k = k + 1; j = 1;
NO
NO
j = j + 1
j = L ?
YES
Magic number - 1
NO
i = 0
Magic number < 0?
YES
Calculate Magic number
x = x + 1
i: 0 ~ C-1, index of effectiveness
j: 1 ~ L, index of voltage level
k: 1~ K, number of nodes in class *c*
j = L ?
j = x;
NO
YES
failed

Fig. 14 Flowchart of multiple voltage scheduling.



Component	Energy (5V)	Energy (3.3V)	Delay (5V)	Delay (3.3V)
Adder	57pJ	25pJ	15.4ns	27.8ns
Multiplier	2202pJ	960pJ	43.7ns	78.9ns

Fig. 15 Energy chart of (5-V,3.3-V) multipliers and adders.



V1-V2	3.3V	5.0V
3.3V	-	178.1pJ
5.0V	61.4pJ	-

Fig. 16 Energy chart of level converters.

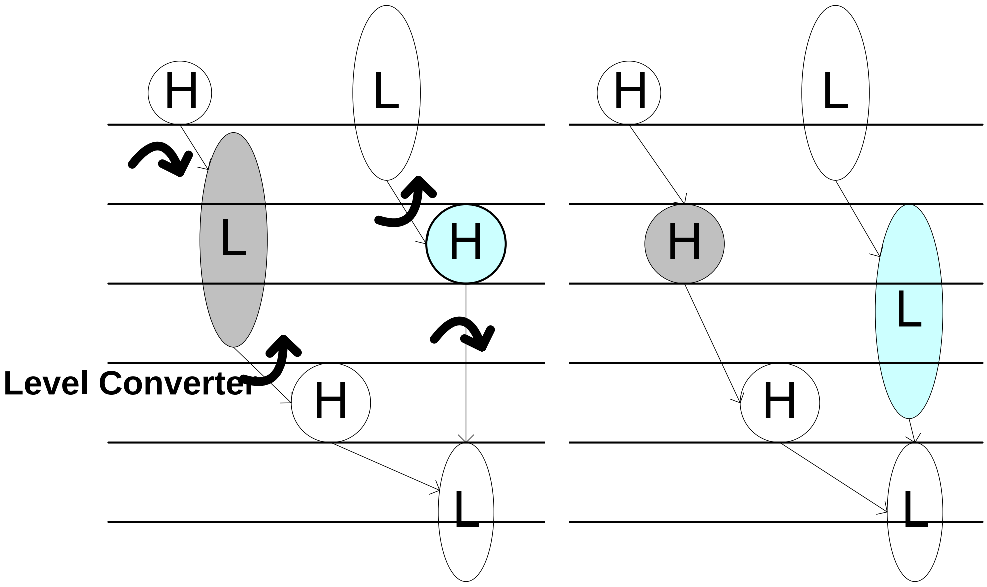


Fig. 17 Examples of level converters.

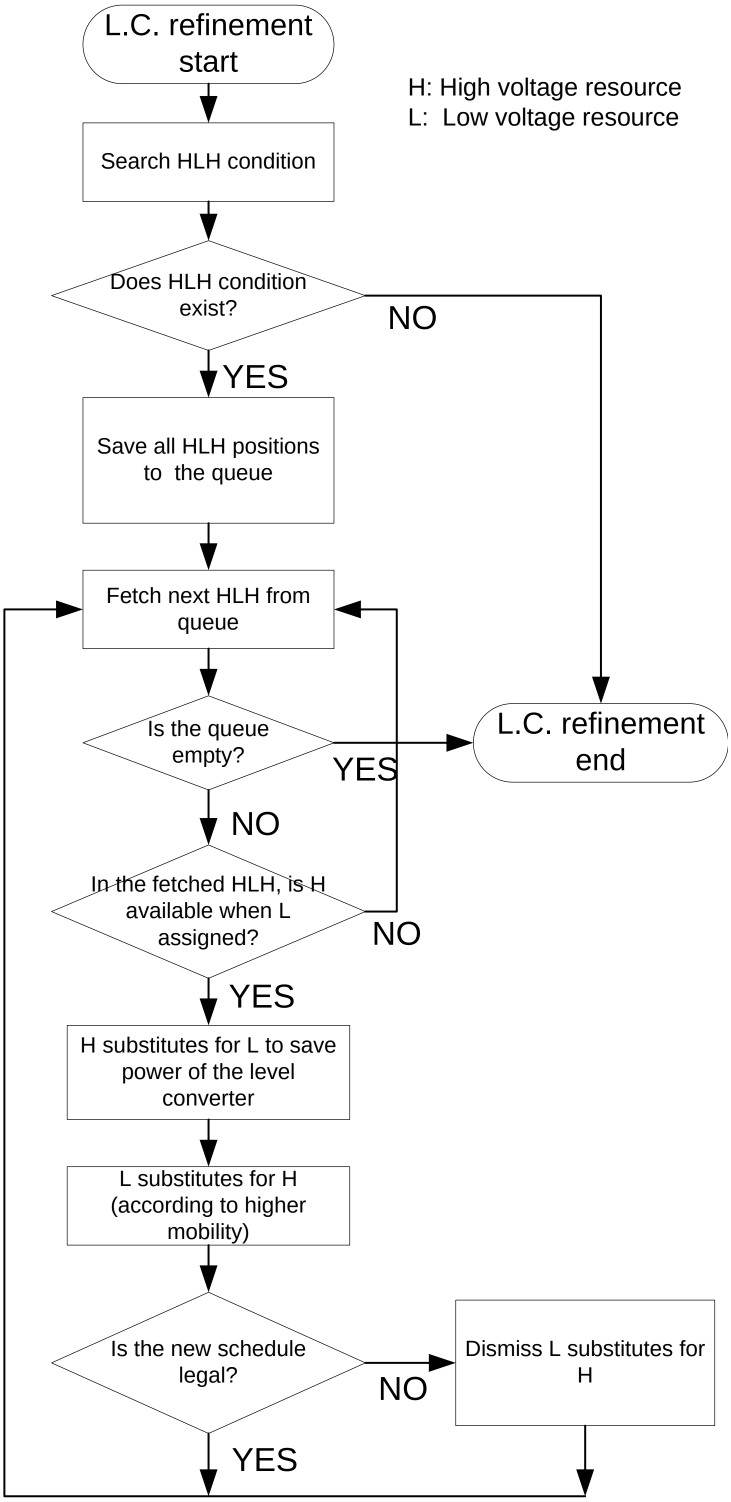

L.C. refinement start
H: High voltage resource
L: Low voltage resource
Search HLH condition
Does HLH condition exist?
NO
YES
Save all HLH positions to the queue
Fetch next HLH from queue
L.C. refinement end
Is the queue empty?
YES
NO
In the fetched HLH, is H available when L assigned?
NO
YES
H substitutes for L to save power of the level converter
L substitutes for H (according to higher mobility)
Is the new schedule legal?
Dismiss L substitutes for H
NO
YES

Fig. 18 Refinement of level converters.

4. Experimental Results

The definition of our multiple-voltage scheduling problem is as follows: given a fixed amount of resources and a specified number of time steps, decide a multiple-voltage scheduling that consumes the minimum energy. The inputs to our algorithm are an FSFG, a resource constraint, and a timing constraint. Resources can be operated at different voltages. We also assume that multiple power lines are available, and level converters are needed between resources if they operate at different voltages. The number of level converters is not user defined. Moreover, the proposed algorithm tries to reduce the number of level converters to save the power consumption. The mobility of each node in the FSFG is defined as the difference between its as-late-as-possible (ALAP) schedule time and its as-soon-as-possible (ASAP) schedule time. Nodes with higher mobility are given more chance during assignment of lower voltage resources. Thereby our proposed algorithm makes use of the loop shrinking transformation and the unfolding transformation to reduce the IPB and to guarantee that the IP meets the IPB in order to get the highest mobilities of each node respectively. The energy consumption and the worst case delays of the different function units adopted from [19] and the energy dissipation of level converters adopted from [9] in this paper are shown in Fig. 15 and Fig. 16. The delay costs of the level converters are absorbed in the worst case delay values. Because we address the problem under timing constraint, energy consumption can be referred as power consumption. We assume the clock period is 20 ns. So the clock cycle of each different function unit can be computed.

We present the results obtained by running our algorithm on some high-level synthesis benchmarks. The algorithm was implemented in C++. Fig. 19 shows the result of a second order IIR filter using proposed approach. The comparison with AR filter (3rd IIR filter) is listed in Fig. 20. In this example, it was found that our algorithm yielded a greater reduction in power consumption. For instance, for the 3rd IIR filter with the resource constraint 1, 1, 1, 1 (one 3.3V multiplier, one 5V multiplier, one 3.3V adder, and one 5V adder) and a timing constraint of 16, we achieve a 40.20% reduction with the unfolding factor P = 3 compared to the 26.00% reduction by using the algorithm in [10]. Fig. 21 describes the power saving results with the specified constraints when running the second order IIR filter and the least mean square adaptive filter examples. The power reduction compared with E_5 has been tabulated in Fig. 22, where E_5 is the power dissipation corresponding to the supply voltage of 5V. E_{alg} is the average power dissipation obtained by our algorithm. T_low is the computation time obtained by application of the minimum-time resource-constrained, i.e., if the latency constraint is less than T_low, a feasible solution cannot be obtained. Timing constraints are given for three different values: T_low, $1.5T_{critical}$, and $2T_{critical}$, where $T_critical$ is the optimal minimum-computation time (critical-path delay) under the given resource constraint. The average reduction obtained by the proposed algorithm with two voltage levels is up to 43.7% when the timing constraint is 1.5 times the critical-path delay.

5. Conclusion

In this paper, we present a new scheduling scheme under resource and latency constraint that minimizes power consumption for the case when the resources operate at multiple voltages. The proposed scheme minimizes the power consumption by assigning as many nodes to lower voltage components as possible by using the algorithmic transformations. The scheme is implemented using the loop shrinking transformation to reduce the IPB and unfolding transformation to guarantee that the IP equals to the IPB and the nodes scheduled using a heuristic-based algorithm and does not guarantee an optimal solution. The average reduction obtained by the proposed algorithm with two voltage levels is 43.7% when the timing constraint is 1.5 times the critical-path delay.

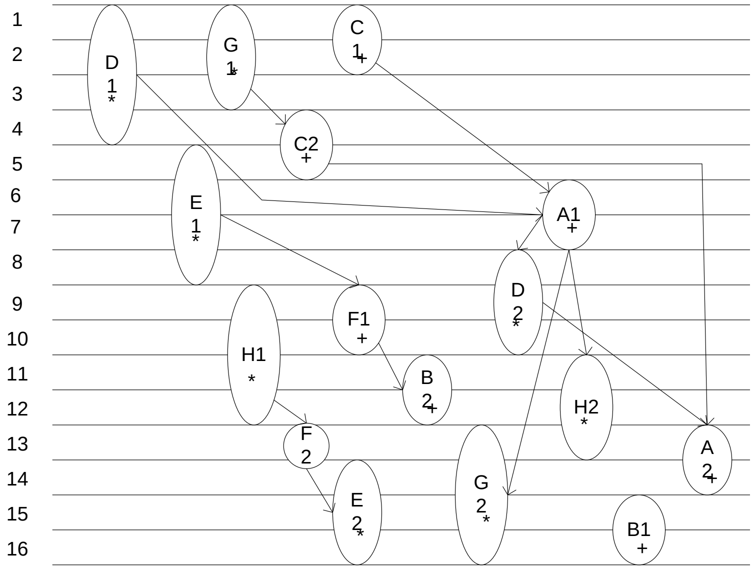
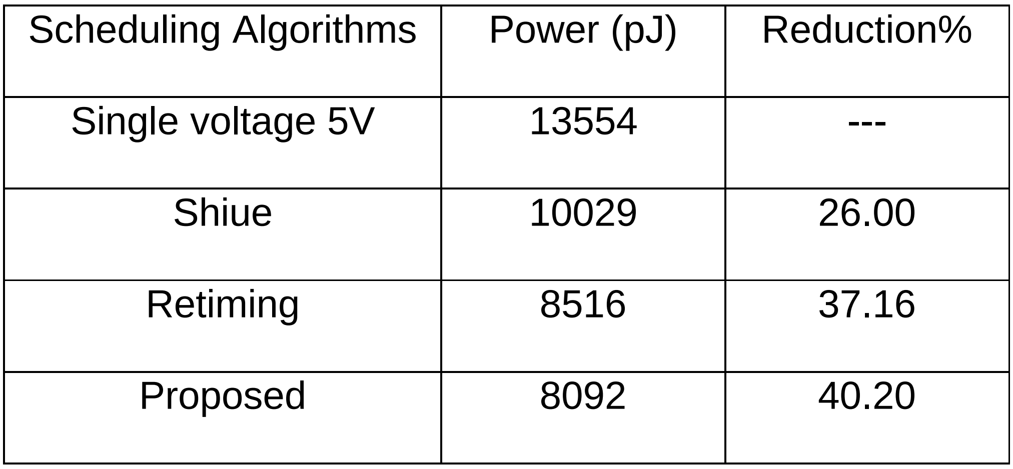


Fig. 19 Experimental result of second-order IIR.



Scheduling Algorithms	Power (pJ)	Reduction%
Single voltage 5V	13554	---
Shiue	10029	26.00
Retiming	8516	37.16
Proposed	8092	40.20

Fig. 20 Comparison results of third-order IIR filter with resource constraint 1, 1, 1, 1 and a timing constraint of 8.

Algorithms	2nd order IIR filter		Least mean square adaptive filter	
	Power (pJ)	Reduction %	Power (pJ)	Reduction %
Single voltage 5V	9036	---	15756	---
Shiue	8208	9.16	13718	12.93
Retiming	6780	24.96	11380	27.77
Proposed	6768	25.10	10768	31.66

Fig. 21 Power consumption and reduction for second-order IIR filter with resource constraint 1, 2, 1, 1, a timing constraint of 8, and unfolding factor of 2 and for LMS filter with resource constraint 1, 1, 1, 1, a timing constraint of 19, and unfolding factor of 2.

Benchmark	Latency	E_{alg} (pJ)	% reduction
2^{nd} IIR filter E_5 = 9039pJ	T_{low}	6768	25.10
	$1.5T_{critical}$	5035	44.28
	$2T_{critical}$	3484	61.44
3^{rd} IIR filter E_5 = 13554pJ	T_{low}	8092	40.20
	$1.5T_{critical}$	6130	54.77
	$2T_{critical}$	5164	61.90
Least mean square adaptive filter E5 = 15756pJ	T_{low}	10768	31.66
	$1.5T_{critical}$	10698	32.10
	2Tcritical	8902	43.50

Fig. 22 Power reduction for the set of benchmarks with resource constraint 1, 1, 1, 1.

References

[1] O.S. Unsal and I. Koren, "System-level power-aware design techniques in real-time systems," Proceedings of the IEEE, vol.91, pp.1055–1069, July 2003.

[2] A.P. Chandrakasan, M. Potkonjak, R. Mehra, J. Rabaey, and R.W. Brodersen, "Optimizing power using transformations," IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, vol.14, pp.12–31, Jan. 1995.

[3] A.P. Chandrakasan, S. Sheng, and R.W. Brodersen, "Low power cmos digital design," IEEE J. Solid-State Circuits, vol.27, pp.473–484, April 1992.

[4] A. Raghunathan and N.K. Jha, "Behavioral synthesis for low power," IEEE Int. Conf. Computer Design: VLSI in Computer and Processors, pp.318–322, Oct. 1994.

[5] A. Raghunathan and N.K. Jha, "An iterative improvement algorithm for low power data path synthesis," IEEE/ACM Int. Conf. Computer-Aided Design, pp.597–602, Nov. 1995.

[6] S. Raje and M. Sarrafzadeh, "Scheduling with two voltages under resource constraints," tech. rep., Dept. Elect. Eng. Comput. Sci., Northwestern Univ. Evanston, IL, 1995.

[7] M. Takahashi, M. Hamada, T. Nishikawa, H. Arakida, T. Fujita, F. Hatori, S. Mita, K. Suzuki, A. Chiba, T. Terazawa, T. Kuroda, and T. Furuyama, "A 60-mw mpeg4 video codec using clustered voltage scaling with variable supply-voltage scheme," IEEE J. Solid-State Circuits, vol.33, pp.1772–1780, Nov. 1998.

[8] M. Sarrafzadeh and S. Raje, "Scheduling with multiple voltages under resource constraints," IEEE Int. Sym. on Circuits and Systems, pp.350–353, May 1999.

[9] J.M. Change and M. Pedram, "Energy minimization using multiple supply voltages," IEEE Trans. VLSI Syst., vol.5, pp.436–443, Dec. 1997.

[10] W.T. Shiue and C. Chakrabarti, "Low power scheduling with resources operating at multiple voltages," IEEE Trans. Circuits and Systems II: Analog and Digital Signal Processing, vol.47, pp.536–543, June 2000.

[11] M.C. Johnson and K. Roy, "Datapath scheduling with multiple supply voltages and level converters," ACM Trans. Design Automation Electronic Syst., pp.227–248, July 1997.

[12] Y.R. Lin, C.T. Hwang, and A.C.H. Wu, "Scheduling techniques for variable voltage low power design," ACM Trans. Design Automation Electronic Syst., pp.81–97, April 1997.

[13] A. Manzak and C. Chakrabarti, "A low power scheduling scheme with resources operating at multiple voltages," IEEE Trans. VLSI Syst., vol.10, pp.6–14, Feb. 2002.

[14] V.K. Madisetti and B.A. Curtis, "A quantitative methodology for rapid prototyping and high-level synthesis of signal processing algorithms," IEEE Trans. Signal Processing, vol.42, no.11, pp.3188–3208, November 1994.

[15] V.K. Madisetti, VLSI Digital Signal Processors, ch. 6, Butterworth-Heinemann, 1995.

[16] T. Barnwell and C. Hodges, "Optimal implementation of signal flow graphs on synchronous multiprocessors," IEEE Int. Conf. Parallel Processing, pp.90–95, August 1982.

[17] K. Parhi and D. Messerschmitt, "Static rate-optimal scheduling of iterative data-flow programs via optimum unfolding," IEEE Trans. Computers, vol.40, pp.178–195, Feb. 1991.

[18] C.T. Hwang, J.H. Lee, and Y.C. Hsu, "A formal approach to the scheduling problem in high level synthesis," IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, vol.10, pp.464–475, April 1991.

[19] S.P. Mohanty, N. Ranganathan, and V. Krishna, "Datapath scheduling using dynamic frequency clocking," IEEE Computer Society Annual Sym. on VLSI, pp.58–63, April 2002.

Lan-Rong Dung was born in 1966. He received a BSEE and the Best Student Award from Feng Chia University, Taiwan, in 1988, an MS in electronics engineering from National Chiao Tung University, Taiwan, in 1990, and Ph.D. in electrical and computer engineering from Georgia Institute of Technology, in 1997. From 1997 to 1999 he was with Rockwell Science Center, Thousand Oaks, CA, as a Member of the Technical Staff. He joined the faculty of National Chiao Tung University, Taiwan in 1999 where he is currently an assistant professor in the Department of Electrical and Control Engineering. He received the VHDL International Outstanding Dissertation Award celebrating in Washington DC in October, 1997. His current research interests include VLSI design, digital signal processing, hardware-software codesign, and System-on-Chip architecture. He is a member of Computer and Signal Processing societies of the IEEE.

Hsueh-Chih Yang was born in 1978. He received the B.S degree in Mechanical Engineering from National Central University, Taoyuan, Taiwan, R.O.C. in 2002. He is currently working toward the Ph.D degree in the Electrical and Control Engineer, National Chiao Tung University. His research interests are power aware system, VLSI architecture, and digital signal processing.

參考文獻

[1] S.-T. Cheng, C.-M. Chen, J.-W. Hwang, “Low-Power Design for Real-Time Systems,” Real-Time Systems, Vol.15, pp.131-148, Kluwer Academic Publishers, 1998.

[2] K. Danckaert, F. Catthoor, H. De Man, “System Level Memory Optimization for Hardware-Software Co-Design,” Proceedi9ngs of the 5th International Workshop on Hardware/Software Co-Design, 1997, pp. 55-64.

[3] L. Benini, A. Macii, E. Macii, M. Poncino, “Selective Instruction Compression for Memory Energy Reduction in Embedded Systems,” International Symposium on Low-Power Electronics and Design, Aug. 1999, pp. 206-211.

[4] L. H. Lee, B. Moyer, J. Arends, “Instruction Fetch Energy Reduction Using Loop Caches for Embedded Applications with Small Tight Loops,” International Symposium on Low-Power Electronics and Design, Aug. 1999, pp. 267-269.

[5] F. Yao, A. Demers, S. Shenker, “A Scheduling Model for Reduced CPU Energy,” IEEE Annual Foundations of Computer Science, pp. 374-382, 1995.

[6] D. Kirovski, M. Potkonjak, “System-Level Synthesis of Low-Power Hard Real-Time Systems,” Design Automation Conference, 1997, pp. 697-702.

［7］ S. Lee, T. Sakurai, “Run-Time Voltage Hopping for Low-Power Real-Time Systems,” DAC’00, June, 2000, pp. 806-809.

［8］ A. Manzak, C. Chakrabarti, “Variable Voltage Task Scheduling for Minimizing Energy or Minimizing Power,” ISLPED’01, 2001, pp. 279-282.

[9] J. Luo, N. K. Jha, "Battery-Aware Static Sheduling for Distributed Real-Time Embedded Systems," DAC’01, 2001, pp. 444-449.

[10] J. Liu, P. H. Chou, N. Bagherzadeh, F. Kurdahi, “Power-Aware Task Motion for Enhancing Dynamic Range of Embedded Systems with Renewable Energy Sources,” Workshop on Power-Aware Computer Systems, PACS’02

[11] “Mobile Pentium III Processor in BGA2 and Micro-PGA2 Packages Datasheet,” Intel Corporation, p55.

[12] K. M. Yang, M T. Sun, and L. Wu, “A family of VLSI designs for the motion compensation block-matching algorithm,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 36, no. 10, pp. 1317-1325, Oct. 1989.

[13] Yeong-Kang Lai, and Liang-Gee Chen, “A data-interlacing architecture

with two-dimensional data-reuse for full-search block-matching algorithm,” *IEEE Trans. Circuits syst. Video Technol.*, Vol. 8, no. 2, pp. 124-127, Apr. 1998.

[14] Chaur-Heh Hsieh, and Ting-Pang Lin, “VLSI Architecture for Block-Matching Motion Estimation Algorithm,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 2, no. 2, pp. 169-175, Jun 1992.

[15] Jen-Chieh Tuan, Tian-Sheuan Chang, and Chein-Wei Jen, “ On the Data Reuse and Memory Bandwidth Analysis for Full-Search Block-Matching VLSI Architecture,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 12, no. 1, pp. 61-72, Jan. 2002.

[16] Mei-Juan Chen, Liang-Gee Chen, and Tzi-Dar Chiueh, “One-dimensional full search motion estimation algorithm for video coding,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 4, no. 5, pp. 504-509, Oct. 1994.

[17] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishigura, “Motion compensated interframe coding for video conferencing,” in *Proc. NTC’81*, New Orleans, LA, pp. G5.3.1-G5.3.5, Nov. 1981.

[18] J. R. Jain, and A. K. Jain, “Displacement measurement and its application in interframe image coding,” IEEE Trans. Commun. Vol. COM-29, pp. 1799-1808, Dec. 1981.

[19] Renxiang Li, Bing Zeng, and Ming L. Liou, “A new three-step search algorithm for block motion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 4, no. 4, pp. 438-442, Aug. 1994.

[20] Ken Sauer, and Brian Schwartz, “Efficient block motion estimation using integral projections,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 6, no. 5, pp. 513-518, Oct. 1996.

[21] Viet L. Do, and Kenneth Y. Yun, “A Low-Power VLSI Architecture for Full-Search Block-Matching Motion Estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 8, no. 4, pp. 393-398, Aug. 1998.

[22] W. Li, and E. Salari, “Succesive elimination algorithm for motion estimation," *IEEE Trans. Image Processing*, Vol. 4, no. 1, pp. 105-107, Jan. 1995.

[23] Wujian Zhang, Runde Zhou, and Kondo, T., “Low-power motion-estimation architecture based on a novel early-jump-out technique,” *The IEEE International Symposium on Circuits and Systems*, Vol. 5 , pp. 187-190, 2001.

[24] Bede Liu, and A. Zaccarin, “New fast algorithms for the estimation of block motion vectors,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 3, no. 2, pp. 148-157, Arp. 1993.

[25] Chok-Kwan Cheung, and Lai-Man Po, “Normalized partial distortion search algorithm for block motion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 10, no. 3, pp. 417-422, Arp. 2000.

[26] Rafael C. Gonzalez, and Richard E. Woods, “Digital Image Processing,” Addison Wesley, Sep. 1993.

計畫成果自評

本計畫第一年已建立功率意識高階合成之基本電池模式，可有助於系統階段合成之設計工作。本計畫之研究成果已發表或已接受發表為下列兩篇期刊論文與一篇會議論文：

1. Hsien-Wen Cheng and Lan-Rong Dung, "A Vario-Power Motion Estimation Architecture Using Content-based Subsample Algorithm," IEEE transactions on Consumer Electronics, Feb. 2004, pp.349-354.
2. Hsien-Wen Cheng and Lan-Rong Dung, "A Power-Aware ME Architecture Using Subsample Algorithm," ISCAS 2004.
3. Lan-Rong Dung and Hsueh-Chih Yang, "On Multiple-Voltage High-Level Synthesis Using Algorithmic Transformations," conditional accepted by IEICE Trans. Fundamentals, 2004.

另外，部分研究成果正投稿于 IEEE 相關期刊。經由本計畫之執行正培養四名博士生及五位碩士生從事於高階合成相關之研究領域。