

行政院國家科學委員會專題研究計畫成果報告

無線通信應用導向 ARM-BASED 內嵌式微處理器系統設計與實作 (3/3)

ARM 處理器指令/程式壓縮/解壓縮之設計與實作

**The Design and Implementation of ARM Processor
Instruction/Program Compression/Decompression**

計畫編號：NSC 90-2215-E-009-107-

執行期限：88 年 8 月 1 日至 91 年 7 月 31 日

主持人：鍾崇斌 教授 國立交通大學資訊工程學系

一、中文摘要

在本年度計劃中，我們探討了如何降低各運算單元的活動，來減少系統的耗能。架構的設計重點包含指令解碼器的加強、運算元選擇單元的建立以及資料路徑的修改等三部分。藉由事先分析各指令的行為，指令解碼器可在指令執行時決定出不需使用之運算單元。透過運算元選擇單元來保持上次運算的值，當運算單元不需運作時，可將上一次運算的值送入運算單元以減少運算單元狀態的轉換所產生的耗能。另外，我們對算術運算邏輯單元和移位器的資料路徑做了小幅度的修改，在發生某些特別運算時，能將運算子適當的繞徑來減少運算單元內部的耗能。

我們在 ARM9TDMI 上面針對此機制進行模擬，比較於原本 ARM9TDMI 上的各運算單元的活動率，模擬結果顯示在算術運算邏輯單元上可節省 20%，移位器可節省 60%，而乘法器則高達 98% 以上，相信本機制在一般嵌入式系統中可以節省處理機內部相當可觀的耗電量。

關鍵詞：低功率設計、嵌入式系統、微處理機架構、訊號轉換

Abstract

In this project, we propose a technique to reduce power consumption by deciding how

and when to turn on/off each function unit to reduce signal switching activities. The proposed method is partitioned to three parts: enhancement of the instruction decoder, Operand-Selection Unit (OSU) design and data-path modification. By profiling instruction behaviors in advance, the instruction decoder can determine the unused function units as early as possible, and then freeze them in the following cycles. The OSU is used to keep the operands in the previous cycle. The number of signal switching activities can be reduced by forwarding the same operands to the function unit in succeeding cycle when this unit will not work in the next cycle. In addition, we modify ALU and shifter data-paths to reduce the power consumption by bypassing some operands when encountering special operations.

We simulate our mechanism on ARM9 TDMI and compare the switching activity for each function unit. Simulation results show that the switching activity reduction of ALU using our mechanism is better 20% than the original, and 60% for shifter. The switching activity reduction of multiplier reaches about 98%.

Keywords: low power design, embedded system, CPU micro-architecture, switching activity

二、緣由與目的

The market of portable devices is growing rapidly, and many applications and challenges appear in the design of embedded system. When the system is getting smaller, low power design becomes more and more important for embedded system design [1].

Observation reveals that function units are clustered in use during program execution, that is, they are sometimes idle for a period of time after serving a burst of computation requests. This provides us an opportunity to design the low power architecture resulting from the unbalanced use of function units. Based on the fact that the switching activities of a CMOS circuit is the major source of power consumption, if we can reduce the number of signal switching activities, the unnecessary power consumption will be saved.

A common concept for reducing power consumption is to turn on/off some functional blocks when not in use. Both clock gating and pre-computation techniques are two main trends of reducing power. Clock gating is the most common technique of reducing power by gating off the clock signals to registers and latches [2]. The pre-computation architecture minimizes switching activity by disabling inputs to the logic circuit [3].

According to the characteristics of ARM instruction set, we could turn on/off each function unit appropriately when the function unit is not in use. When a result of a function unit can be ignored, the power can be saved

by preventing irrelevant switching activity caused by the computation of unused data. In this project, we will investigate instruction behavior in detail and design related mechanism to handle unused function unit

三、結果與討論

Architecture Overview

Based on pre-computation architecture [3], the proposed micro-architecture is designed in the decoding (ID) and execution (EX) stages of the ARM9TDMI processor core. In the ID stage, the decoder generates control signals for each functional block to achieve instruction execution according to instruction types. We utilize the decoding function to make unused function unit idle. A new function block named *Partial-Latch-Control* (PLC) unit is added into the decode stage, as shown in Figure 1. PLC retains partial control codes or/and operand values operated in the previous cycle in ID/EX inter-stage latch for the purpose that when the decoder indicates some function units are not used in the following execution of the current instruction. The latched control codes or operand values can keep the internal states of function units unchanged, thus, reduces the signal switching activates.

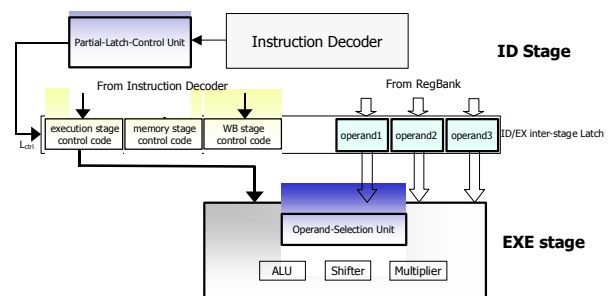


Figure 1. Architecture Overview

In the EX stage, we divide the design into

two parts to solve the power reduction problem. One is to build *Operand-Selection Unit* (OSU) and the other is to construct extra data-paths. In order to save power consumption caused by switching activity of function units, we freeze the function unit by keeping both the control signals and operand values as the same ones in the previous cycle. The OS unit is proposed to keep each operand value unchanged for function unit (while the control codes are preserved by the PLC unit) as shown in Figure 2. Each OSU consists of a data latch and a multiplexer as shown in Figure 3. The data latch is used to store the operand of current operation for further use, and the multiplexer is used to select which one (the latched previous operand or current operand) being operated for the function unit.

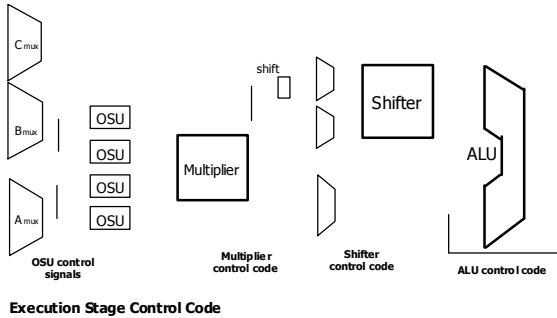


Figure 2. The modified ARM9TDMI

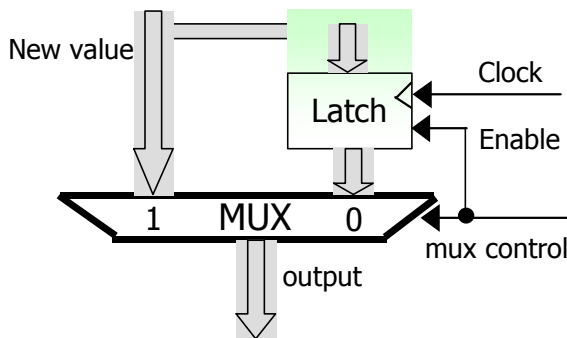


Figure 3. Operand-Selection Unit Design

Extra data-paths are used to solve the problem that multiple function units share the same input bus in ARM9TDMI organization.

Sharing buses is efficient for low cost, but it is not necessary for low power consumption. The extra data-paths are proposed to aim at ALU and shifter. We find the operations ‘move’ and ‘shift’ with an immediate shift amount of zero are redundant since these operations only transfer the contents of a register to another. We call these operations dummy operations. According to the processor organization, the contents of the source register will pass through the function unit to the destination register because of the sharing buses. We route these operands that need not operation bypassing the function units to reduce the power consumption. By analyzing the program behavior, we find that the number of ‘shift’ operations with an immediate shift amount of zero occupies one third of total shift operations and the number of move operation is about 16% of the total ALU operations.

3.4 Experimental Results

We use MediaBench [4] benchmark programs to evaluate the switching activity using our mechanism, and then compare the switching activity with that of ARM9TDMI. Figure 4 shows the results. The X-axis is the selected benchmarks. The Y-axis is the normalized numbers of switching activities reduced of the function units of ARM9TDMI. The statistics of ALU, shifter and multiplier are collected. Each benchmark has three bars, indicating the normalized numbers of switching activities of three function units, respectively. Each bar consists of two parts: the blue bar indicates the number of activities applying freezing mechanism (using both OSU and PLU) and the red bar indicates the

number of activities counting for dummy instructions. In this figure, we find the reduction for multiplier is the most, because the multiplier is seldom used. The power is reduced by the largest degree. On the other hand, it is worth paying more attention that the reduction for ALU is the least, since: (a) ALU operates most of time and (b) the number of dummy instruction is rare. ALU operates for most of instructions, including load/store, data processing and branches. There is not much opportunity to reduce the signal switching activities of ALU. Figure 4 also shows that the reduction effect of dummy operations is much better than that of freezing mechanism for both ALU and shifter, because the number of dummy operations is much more than the situations when the function unit is frozen. After all, the reduction of switching activity of function unit is notable adopting the proposed method.

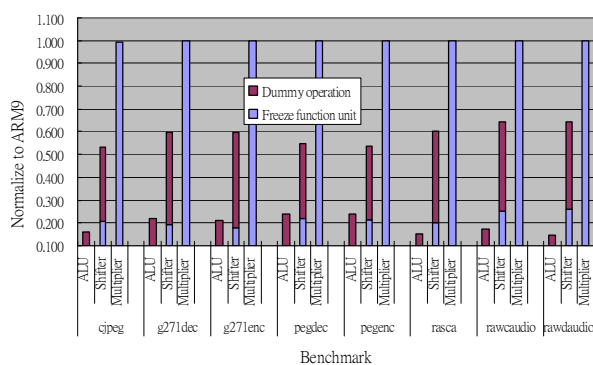


Figure 4. Reduction of Switching Activity

四、計畫結果自評

With freezing function units and handling dummy operations, a modified low power ARM9TDMI micro-architecture is proposed. The key idea of our mechanism is to elastically disable the inputs of a combinational circuit according to the instruction characteristics during program

execution. Different from clock gating method, this method supports a low power design for combinational circuit without large critical path delay. Combining clock gating and this method can achieve much more power reduction in most processor design.

五、參考文獻

- [1]. Moyer, B., "Low-power design for embedded processors", *Proceedings of the IEEE*, vol. 11, pp. 1576-1587, Nov. 2001.
- [2]. Qing Wu Pedram, M. Xunwei Wu, "Clock-gating and its application to low power design of sequential circuits", *IEEE Trans. Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, pp. 415-420, Mar 2000.
- [3]. M. Alidina, J. Monteriro, S. Devadas, A. Ghosh and M. Papaefthymiou, "Precomputation-Based Sequential Logic Optimization for Lower Power," *IEEE Trans. VLSI Syst.*, vol. 2, pp. 426-436, Dec. 1994.
- [4]. C. Lee, M. Potkonjak, and W. H. M.-Smith, "MediaBench: A Tool for Evaluating and Synthesizing Multimedia and Communications Systems", *30th Annual ACM/IEEE International Symposium on Microarchitecture*, 1997.
- [5]. <http://www.cs.ucla.edu/~leec/mediabench/applications.htm>
- [6]. *Advanced RISC Machines Ltd.*, "ARM Architecture Reference Manual", July 1996.
- [7]. *Advanced RISC Machines Ltd.*, "ARM 9TDMI Data Sheet", 2000.

