An Operation Rearrangement Technique for Low-Power VLIW Instruction Fetch

Dongkun Shin* and Jihong Kim

Computer Architecture Lab
School of Computer Science and Engineering
Seoul National University, Korea
Outline

- Motivations
- VLIW Instruction Encodings
- LOR Problem and Solution
- GOR Problem and Solution
- Experiment
- Conclusions
Motivations

Many mobile devices are designed using VLIW processors for high performance, which usually consume more power than single-issue processors.

In digital CMOS circuits, switching activity accounts for over 90% of total power consumption.

We propose a post-pass optimization technique that can reduce switching activity during the instruction fetch phase in VLIW processors.
VLIW Instruction Encoding-Uncompressed

Program

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Functional Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>IADD</td>
<td>IntU</td>
</tr>
<tr>
<td>FADD</td>
<td>IntU</td>
</tr>
<tr>
<td>LOAD</td>
<td>IntU</td>
</tr>
<tr>
<td>STORE</td>
<td>IntU</td>
</tr>
<tr>
<td>ISUB</td>
<td>IntU</td>
</tr>
<tr>
<td>IMUL</td>
<td>IntU</td>
</tr>
<tr>
<td>IADD</td>
<td>IntU</td>
</tr>
<tr>
<td>BEG</td>
<td>BrU</td>
</tr>
</tbody>
</table>

Alternative encoding
VLIW Instruction Encoding - Compressed

Possible choices = 4!

Which encoding is the best for low-power consumption?
Machine Model

Memory block is fetched from the main memory through the $b_{\text{mem}}$-bit width instruction bus on cache-miss.

Because of the compressed encoding format, several VLIW instructions are fetched together in a single fetch from the instruction cache.

A fetch packet consists of $N$ operations, and $b_{\text{mem}} = b_{\text{cache}}/N$
Basic Idea

Instruction Cache

(a) Before operation rearrangement

(b) After operation rearrangement

The total # of bit changes are reduced by 25%
Problem Formulation

Problem:
how to reorder given VLIW instructions to reduce the number of bit transitions between successive instruction fetches.

Solutions:
Local Operation Rearrangement (LOR): each basic block is independently considered.
Global Operation Rearrangement (GOR): all the basic blocks are simultaneously considered.
LOR Problem

$$SW^B = SW_{cache}^B + \alpha \cdot SW_{mem}^B$$

$\alpha$ is the load capacitance ratio of the external instruction bus to the internal instruction bus.

$SW_{cache}^B$ is the number of bit changes at the internal instruction bus.

$SW_{mem}^B$ is the number of bit changes at the external instruction bus.
The LOR Problem involves optimizing the flow of data between different memory levels and between processor core and external memory. The equation for calculating the total switch cost ($SW^B$) is expressed as:

$$SW^B = \sum_{FP} SW^{\text{intra}}_{FP} + \sum_{FP} SW^{\text{inter}}_{FP}$$

Where $SW^{\text{intra}}_{FP}$ represents the switch cost within the memory hierarchy and $SW^{\text{inter}}_{FP}$ represents the switch cost between memory levels and the processor core.
Solution for LOR

$EQ(FP_i^B)$ : The set of equivalent fetch packets of $FP_i^B$.
Solution for LOR

- We find the **shortest path** from START to END, which is the solution of operation rearrangement to minimize the SW^B.

- A node v_{i+1} in graph finds the node v_i through which the shortest path from START to the node v_{i+1} should pass.
GOR Problem

- All the basic blocks in a program are simultaneously considered
  - how many times each basic block is executed.
  - how often each basic block experiences cache misses.
  - how basic blocks are related each other.

\[ SWS = \sum_{BB} \sum_{BB} SW_{inter}(bb_i, bb_j) + \sum_{BB} SW_{intra}(bb_i) \]

- \( SW_{inter} \) and \( SW_{intra} \) is represented by \( SW_{FP} \), weight of each basic block, and cache miss rate.
Solution for GOR

This method may require an excessive amount of memory and cycles.

We need a heuristic solution.
Heuristic for GOR

- All the basic blocks are not equally treated.
  - Basic blocks with larger effects on the total switching activity are more thoroughly reordered than ones with smaller effects.
- Not all the equivalent basic blocks in $EQ(bb_i)$ are tried to find an optimal solution.
  - Only $N_{cand}$ equivalent basic blocks are created and included in graph.
Fixed-point DSP

VLIW processor that can specify eight 32-bit operations in a single 256-bit instruction.

Use a compressed encoding
For our benchmark programs, the bit transitions was reduced by 34% on an average.
Conclusions

- Described a post-pass optimal operation rearrangement method for low-power VLIW instruction fetch.
  - The switching activity was reduced by 34% on average.

- Future works
  - The phase-ordering problem between the operation rearrangement and other compiler optimization steps.
  - Operation rearrangement problem in super-scalar processors.