Performance and Power Consumption Analysis of Arm Scalable Vector Extension with gem5 Simulator

Tetsuya Odajima, Yuetsu Kodama, Miwako Tsuji, Mitsuhisa Sato
RIKEN R-CCS, Japan
Background of Our Research

• We have been working on the performance evaluation and application tuning for A64FX processor of Post-K by the simulator

• Arm Scalable Vector Extension (SVE)
  • Support from 128-bit to 2048-bit vector length (VL)
  • Vector Length Agnostic (VLA) ; there is no VL information in assembly code

• Working on research of processor architecture by using simulator
  • Simulator can change its various hardware parameters easier
  • We use alternative parameters, not that of the Post-K

• Our simulator environment
  • gem5-sve: general-purpose processor simulator, supports SVE
    • We have been implementing some instructions
  • McPAT: framework for calculating power consumption
    • Estimate the power consumption from the # of access to FPU, registers, caches, etc.
Our Simulator Environment

- Flow of estimating area and power consumption

Applications -> gem5-sve --> McPAT

- Statistical information
  - # of access to FPU
  - Cache hit ratio
  - Hardware configurations

- Process rule
- ITRS information

Calculate power consumption from reference energy consumption of FPU and access frequency
Purpose of Our Research

• We will evaluate the effect of vector length in the performance and the energy consumption by using simulator

• Comparing the performance and energy consumption among 512-bit and 1024-bit vector length
  • Peak performance of 1024-bit FPU is twice that of 512-bit
    → Area of FPUs and register fires also will be increasing nearly twice

• Various implementations of FPU

<table>
<thead>
<tr>
<th>512bit Vector</th>
<th>1 cycle issue</th>
<th>1024bit Vector</th>
<th>1 cycle issue</th>
<th>1024bit Vector</th>
<th>2 cycles issue</th>
</tr>
</thead>
<tbody>
<tr>
<td>512bit FPU</td>
<td></td>
<td>1024bit FPU</td>
<td></td>
<td>512bit FPU</td>
<td></td>
</tr>
</tbody>
</table>

• We will evaluate the advantage of a longer vector length with multi-cycle FPU

Peak performance is half
\[ \uparrow \text{trade-off} \]
Area of FPU is also half
Evaluation Environment

• Based on ThunderX2 parameter
• We defined instruction latency for SVE referred to NEON

<table>
<thead>
<tr>
<th>Hardware parameters</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock Frequency</td>
<td>2.1 GHz</td>
<td></td>
</tr>
<tr>
<td># of cores</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>L1 Dcache, Icache size</td>
<td>32 kB</td>
<td>L2 cache size</td>
</tr>
<tr>
<td>Integer pipeline</td>
<td>1</td>
<td>Load/Store unit</td>
</tr>
<tr>
<td>Floating pipeline</td>
<td>2</td>
<td>Process rule</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Out-of-Order resource parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>IQ (Reservation Station)</td>
</tr>
<tr>
<td>ROB (Re-order Buffer)</td>
</tr>
<tr>
<td>LQ (Load Queue)</td>
</tr>
<tr>
<td>SQ (Store Queue)</td>
</tr>
<tr>
<td>Physical Vector Register</td>
</tr>
</tbody>
</table>
Evaluation Environment (Cont’d)

• Compiler
  • gcc-arm 8.2, -Ofast -march=armv8-a+sve

• Kernels
  • N-body: latency-bound
    • N=512, TIME_STEP=1
  • Stream Triad: cache/memory-bound
    • N=25600, TIME_STEP=10

• In this evaluation, we don’t consider “leak current”
Evaluation Environment (Cont’d)

• In gem5, SIMD width of execution unit and register file will be changed by the VL

<table>
<thead>
<tr>
<th></th>
<th>LEN=4</th>
<th>LEN=8 half</th>
<th>LEN=8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vector Length</td>
<td>512-bit</td>
<td>1024-bit</td>
<td></td>
</tr>
<tr>
<td>FPU throughput</td>
<td>512-bit / cycle</td>
<td>1024-bit / 2cycles</td>
<td>1024-bit / cycle</td>
</tr>
<tr>
<td>Peak FPU</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>performance ratio</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of registers</td>
<td>512-bit x 144</td>
<td>1024-bit x 144</td>
<td></td>
</tr>
</tbody>
</table>

Diagram:

- 512bit Vector
  - 512bit FPU
  - 1 cycle issue
- 1024bit Vector
  - 512bit FPU
  - 2 cycles issue
- 1024bit Vector
  - 1024bit FPU
  - 1 cycle issue
Hardware Area

- Result of McPAT
- Focus on only FPU & FP Reg. files
- Comparing to LEN=4,
  - area of LEN=8 is +80%
  - area of LEN=8 half is only +14%

- The impact to multi-core is great
Evaluation | Time & Energy Consumption

N-body

Stream Triad

![Bar charts comparing energy consumption and execution time for N-body and Stream Triad benchmarks.](image)

- N-body
  - LEN=4
  - LEN=8 half
  - LEN=8

- Stream Triad
  - LEN=4
  - LEN=8 half
  - LEN=8

Better

# of physical vector registers == 144

Runtime dynamic power [W]

Execution time [msec]

Dynamic runtime power [W]
Conclusion

- By increasing the vector length, performance improvement and low power consumption can be realized in latency-bound kernel
  - N-body: ”LEN=8 half” achieves 17% speedup than LEN=4
    5% low energy

- Vector length dose not affect in the performance of memory-bound kernel

- The energy consumption of “LEN=8 half” is almost the same as that of LEN=8

- A longer vector length with multi-cycle vector units (LEN=8 half) is well balanced between the performance and hardware resource
Download Our Simulator

- We provide Docker image includes our simulator and open source compiler for SVE