Introduction

- On-chip parallel computation shows great promise for scaling raw processing performance.
- Chip multiprocessors (CMPs) often struggle with programmability and scalability issues (cache coherency and off-chip memory bandwidth).
- Programming a multiprocessor system except parallelism requires: mapping threads to processors, distributing data to optimize locality etc.
- On-chip processing unit is busy due to programability issues.
The SARC architecture

• Is based on a heterogeneous set of processors managed at runtime in a master-worker mode.

• Due to ability to schedule data ahead in time applications tolerate long memory latencies.

• Focus: sufficient bandwidth to feed data to all workers.

• SARC architecture’s potential has a broad range of parallel computing scenarios (multimedia, bioinformatics, and scientific domains).
Programming model

• New class of task-based data-flow programming models that includes: StarScs, Cilk, RapidMind, Sequoia, and OpenMP 3.0.

• StarScs consists of:
  - a source-to-source compiler
  - supporting runtime library
An asymmetric chip multiprocessor

Worker P: feature a local scratchpad memory. Mapped: app. logical address space. Access: through L/S

Master: Starting up the application at the program’s main subroutine. Access through cache

NoC = K-bus org
*no previous request: node is dynamically assigned one of the buses.

*Captures both misses from the L1 caches and DMA transfers;

Issue requests to the DRAMs: FIFO Requests handle in order, but can execute out of order with respect to requests sent to another channel

A few high-performance master processors and clusters of worker processors that are customized to a target application domain.
The SARC worker processors are based on different designs, depending on application domain.

- **Media accelerator**: is an application-specific instruction set processor (ASIP) based on the Cell instruction set processor (ASIP) based on the Cell synergistic processor element (SPE).

  **Goal**: Adding few application-specific instructions can achieve significant performance improvements.

(a)  
<table>
<thead>
<tr>
<th></th>
<th>Speedup</th>
<th>Instruction count reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDCT8</td>
<td>2.5</td>
<td>2.0</td>
</tr>
<tr>
<td>IDCT4</td>
<td>3.0</td>
<td>2.5</td>
</tr>
<tr>
<td>DF</td>
<td>2.0</td>
<td>1.5</td>
</tr>
<tr>
<td>Luma</td>
<td>2.5</td>
<td>2.0</td>
</tr>
</tbody>
</table>
- **Bioinformatics accelerator:**
  - 3 new instructions (Max instruction, two instructions to speed up Smith-Waterman formula)
  - 17% improve

- **Scientific vector accelerator**
  Register file lets us define 1D and 2D logical vector registers of different sizes and shapes.
Performance analysis

- TaskSim: provides cycle-accurate simulation of the entire SARC architecture, including the DMAs, caches, MICs, and DRAMs.
- TaskSim is highly scalable because workers need not be simulated at the instruction level, therefore accurately simulating workers by only simulating their DMA transfers and task synchronizations.
- Intertask dependency information: changing the number of simulated processors does not break application semantics.

Table 1. Baseline SARC simulation parameters.

Table 2. Characteristics of the SARC applications.

<table>
<thead>
<tr>
<th>Application</th>
<th>No. of tasks</th>
<th>Task duration (microseconds)</th>
<th>Bandwidth per task (GBps)</th>
<th>Problem size (Mbytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>H.264</td>
<td>816,000</td>
<td>17.4</td>
<td>0.65</td>
<td>299</td>
</tr>
<tr>
<td>Smith-Waterman</td>
<td>3,670,016</td>
<td>50.3</td>
<td>0.65</td>
<td>20.7</td>
</tr>
<tr>
<td>Matrix multiply</td>
<td>262,144</td>
<td>25.8</td>
<td>1.42</td>
<td>192</td>
</tr>
<tr>
<td>Cholesky</td>
<td>357,760</td>
<td>28.0</td>
<td>1.68</td>
<td>512</td>
</tr>
</tbody>
</table>
Parallel scalability

- Performance improvement by increasing the number of workers.

Impact of memory latency

Performance degradation of the target application by increasing latency.
Impact of memory bandwidth

- MatrixM=1,42 bandwidth
- Pin count prevent providing bandwidth-L2 helps

Performance degrades as we reduced L2 cache size

So we conclude that distributed on-chip L2 cache can effectively filter the off-chip memory bandwidth
Impact of specialization

SARC architecture’s scalability using the SARC domain specific accelerators instead of the baseline worker processor.

Combined impact of scalability and accelerators.

Smith Waterman: the accelerator is 17% faster than base processor.
H264-2D: don’t suffer any scalability impact.
H264-3D: bandwidth is enough to sustain accelerators core scalability, but has 11% reduction on speedup.
Conclusion

• The SARC architecture offers scalability with the number of workers and combines well with an heterogeneous set of domain-specific accelerators to achieve the desired performance level at a lower cost and power.

• http://www.sarc-ip.org/index.jsp?p=1