Verification of Transactional Memory in POWER8™

Allon Adir, Dave Goodman, Daniel Hershcovich, Oz Hershkovitz, Bryan Hickerson, Karen Holtz, Wisam Kadry, Anatoly Koyfman, John Ludden, Charles Meissner, Amir Nahir, Randall R Pratt, Mike Schiffli, Brett St Onge, Brian Thompto, Elena Tsanko, Avi Ziv
POWER8™ Processor

Technology
- 22nm SOI, eDRAM, 15 level metal
- 650mm²

Cores
- 12 cores (SMT8)
- 8 dispatch, 10 issue, 16 exec pipe
- 2X internal data flows/queues
- Enhanced prefetching
- 64K data cache
- 32K instruction cache

Accelerators
- Crypto and memory expansion
- **Transactional Memory**
- VMM assist
- Data Move / VM Mobility

Caches
- 512 KB SRAM L2 / core
- 96 MB eDRAM shared L3
  - (low latency local 8M regions)
- Up to 128 MB eDRAM L4 (off-chip)

Memory
- Up to 230 GB/s sust. Bandwidth

Energy Mgmt
- On-chip Mgmt and VRM, Gating
Locking vs. Transactional Memory

deposit(acct, amt){
  lock(&acct);
  curr = acct->balance;
  new = curr + amt;
  acct->balance = new;
  unlock(&acct);
}

deposit(acct, amt){
  transaction {
    curr = acct->balance;
    new = curr + amt;
    acct->balance = new;
  }
}

T1 T2 T3

T1 T2 T3

T1 T2 T3
Transaction Pseudocode

// atomic_inc
start:
  tbegin.
  bc- tm_fail_handler

crit_sec:
  lw r1, 0(r3)
  addi r1, r1, 1
  stw r1, 0(r3)
  tend.

tm_fail_handler:
  // Examine failure cause
  mftexasr
  // Retry N times
  bc start
  // get a lock
  bc crit_sec
<table>
<thead>
<tr>
<th>Thread 1</th>
<th>Thread 2</th>
<th>Thread 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>bc-</td>
<td>bc-</td>
<td>bc-</td>
</tr>
<tr>
<td>stb x11 \rightarrow x5000</td>
<td>stb x22 \rightarrow x5000</td>
<td>ibz G1 \leftarrow x5000</td>
</tr>
<tr>
<td>Q 5000 11</td>
<td>Q 5000 22</td>
<td>R G1 00 *IN</td>
</tr>
<tr>
<td>stb x11 \rightarrow x6000</td>
<td>stb x22 \rightarrow 0x6000</td>
<td>R G1 11 *T1</td>
</tr>
<tr>
<td>Q 6000 11</td>
<td>Q 6000 22</td>
<td>R G1 22 *T2</td>
</tr>
<tr>
<td>tend</td>
<td>tend</td>
<td>tend</td>
</tr>
<tr>
<td>/* Q \rightarrow D</td>
<td>/* Q \rightarrow D</td>
<td>/* Q \rightarrow D</td>
</tr>
<tr>
<td>D 5000 11</td>
<td>D 5000 22</td>
<td>R G2 00 *IN</td>
</tr>
<tr>
<td>D 6000 11</td>
<td>D 6000 22</td>
<td>R G2 11 *T1</td>
</tr>
</tbody>
</table>

T3 must see either: G1 00 and G2 00 *IN
   G1 11 and G2 11 *T1
   G1 22 and G2 22 *T2
Non-determinism

- Verification Engineers like deterministic behavior
  \[2 + 2 = 4\]

- TM is NOT deterministic!!
  - a transaction might:
    - commit successfully
    - fail (for lots of reasons)
  
  - Relies on hardware and software working together

  - Not verification friendly!!
Footprint Size Effects Successful Commit Probability

- small footprint implies high success rate
- hard to predict outcome
- very large footprint guarantees failure
Transaction Passing Rate Affected by Shared Footprint Capacity (approximation)

Number of Lines in Footprint

Transaction Passing Percentage

Threads per Core
Suspend Mode

- Pause transaction and then resume
  - Interrupts
  - Flag setting
  - Software debug

- Loads and stores performed non-transactionally
  - Does not pollute footprint
  - Stores immediately visible to other threads

- Hardware still tracks footprint collisions
  - Failure handling deferred until transaction resumed
**Suspend Mode**

**Initial Memory:**

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Thread 1</td>
<td>Thread 2</td>
<td></td>
</tr>
<tr>
<td>tbegin *T1</td>
<td>tbegin *T2</td>
<td></td>
</tr>
<tr>
<td>bc-</td>
<td>bc-</td>
<td></td>
</tr>
<tr>
<td>stb x11 (\rightarrow) x5000</td>
<td>lbz G1 (\leftarrow) x5000</td>
<td></td>
</tr>
<tr>
<td>Q 5000 11</td>
<td>R G1 00 *IN</td>
<td></td>
</tr>
<tr>
<td>tsuspend</td>
<td></td>
<td></td>
</tr>
<tr>
<td>stb x22 (\rightarrow) x6000</td>
<td>lbz G2 (\leftarrow) x6000</td>
<td></td>
</tr>
<tr>
<td>D 6000 22</td>
<td>R G2 00 *IN</td>
<td></td>
</tr>
<tr>
<td>tresume</td>
<td></td>
<td></td>
</tr>
<tr>
<td>stb x33 (\rightarrow) x5000</td>
<td>R G2 22 *T1 (suspend store)</td>
<td></td>
</tr>
<tr>
<td>Q 5000 33</td>
<td>tend</td>
<td></td>
</tr>
<tr>
<td>transaction failure occurs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(for any reason)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/* Discard Q values, Keep D values</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D 5000 00</td>
<td>G1 00 *IN &amp; G2 00 *IN</td>
<td></td>
</tr>
<tr>
<td>D 6000 22</td>
<td>G1 00 *IN &amp; G2 22 *T1</td>
<td></td>
</tr>
</tbody>
</table>

T2 must see either:

- G1 00 *IN & G2 00 *IN
- G1 00 *IN & G2 22 *T1
Large Verification Methodology Impact

Percentage of Transactional Memory Bugs Found by Platform

- Block: 16%
- Unit: 4%
- Element: 8%
- System: 2%
- Review: 2%
- Accelerator Platform: 2%
- Post-Silicon: 1%
- Total: 67%
Thank You!