Representative Litmus Tests

Jade Alglave and Luc Maranget
The contents of this document are provided in connection with the HSA Foundation specifications. This specification is protected by copyright laws and contains material proprietary to the HSA Foundation. It or any components may not be reproduced, republished, distributed, transmitted, displayed, broadcast or otherwise exploited in any manner without the express prior written permission of HSA Foundation. You may use this specification for implementing the functionality therein, without altering or removing any trademark, copyright or other notice from the specification, but the receipt or possession of this specification does not convey any rights to reproduce, disclose, or distribute its contents, or to manufacture, use, or sell anything that it may describe, in whole or in part.

HSA Foundation grants express permission to any current Founder, Promoter, Supporter Contributor, Academic or Associate member of HSA Foundation to copy and redistribute UNMODIFIED versions of this specification in any fashion, provided that NO CHARGE is made for the specification and the latest available update of the specification for any version of the API is used whenever possible. Such distributed specification may be re- formatted AS LONG AS the contents of the specification are not changed in any way. The specification may be incorporated into a product that is sold as long as such product includes significant independent work developed by the seller. A link to the current version of this specification on the HSA Foundation web-site should be included whenever possible with specification distributions.

HSA Foundation makes no, and expressly disclaims any, representations or warranties, express or implied, regarding this specification, including, without limitation, any implied warranties of merchantability or fitness for a particular purpose or non-infringement of any intellectual property. HSA Foundation makes no, and expressly disclaims any, warranties, express or implied, regarding the correctness, accuracy, completeness, timeliness, and reliability of the specification. Under no circumstances will the HSA Foundation, or any of its Founders, Promoters, Supporters, Academic, Contributors, and Associates members or their respective partners, officers, directors, employees, agents or representatives be liable for any damages, whether direct, indirect, special or consequential damages for lost revenues, lost profits, or otherwise, arising from or in connection with these materials.
Acknowledgments

This document is the result of the contributions of many people. Here is a partial list of the contributors, including the companies that they represented at the time of their contribution.

INRIA

- Luc Maranget

Microsoft Research Cambridge and University College London

- Jade Alglave
# Contents

Acknowledgments .................................................................................................. 3

About this document: Representative Litmus Tests ........................................... 6

   Audience ........................................................................................................... 6
   HSA Information Sources ................................................................................ 6

1. Sequentially-consistent synchronizing operations ........................................... 7
2. Synchronization between units of execution ......................................................... 8
3. Transitivity with a single scope .......................................................................... 9
4. Synchronization through multiple scopes .......................................................... 10
5. Synchronization through scope inclusion .......................................................... 11
6. Synchronization through scope inclusion and scope transitivity ....................... 12
7. Consistency required for coh and hhb ............................................................. 13
8. Synchronization between units of execution using relaxed atomics ............... 15
9. Unobservable store speculation ....................................................................... 16
10. No out-of-thin-air values .................................................................................. 17
11. Dekker’s algorithm ......................................................................................... 18
12. Conflict without synchronization .................................................................... 19
13. Insufficient scope ........................................................................................... 20

## Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-1</td>
<td>HSA01 is forbidden by the consistency of SCwg → coh → (Forbidden)</td>
<td>7</td>
</tr>
<tr>
<td>2-1</td>
<td>HSA02 is forbidden by the consistency of hhb → coh → (Forbidden)</td>
<td>8</td>
</tr>
<tr>
<td>3-1</td>
<td>HSA03 is forbidden by the consistency of hhb → coh → (Forbidden)</td>
<td>9</td>
</tr>
<tr>
<td>4-1</td>
<td>HSA04 is forbidden by the consistency of hhb → coh → (Forbidden)</td>
<td>10</td>
</tr>
<tr>
<td>5-1</td>
<td>HSA05 is forbidden by the consistency of hhb → coh → (Forbidden)</td>
<td>11</td>
</tr>
<tr>
<td>6-1</td>
<td>HSA06 is forbidden by the consistency of hhb → coh → (Forbidden)</td>
<td>12</td>
</tr>
<tr>
<td>7-1</td>
<td>Test HSA07 is forbidden by the consistency of hhb → coh →</td>
<td>13</td>
</tr>
<tr>
<td>7-2</td>
<td>HSA07 is forbidden by value-of-a-load rule</td>
<td>14</td>
</tr>
<tr>
<td>8-1</td>
<td>HSA08 is forbidden by the consistency of hhb → coh → (Forbidden)</td>
<td>15</td>
</tr>
<tr>
<td>9-1</td>
<td>HSA09 is forbidden by the irreflexivity of gdo → (ldo → ∪ rfe →)</td>
<td>16</td>
</tr>
<tr>
<td>10-1</td>
<td>Test HSA10 is forbidden by the irreflexivity of gdo → (ldo → ∪ rfe →)</td>
<td>17</td>
</tr>
<tr>
<td>11-1</td>
<td>Test HSA11 is allowed</td>
<td>18</td>
</tr>
</tbody>
</table>
About this document: Representative Litmus Tests

This document contains examples, illustrations of forbidden executions, and links to web interfaces of the herd simulator of the HSA model.

Audience

This document is written for system and component architects interested in supporting the HSA infrastructure (hardware and software) within platform designs.

HSA Information Sources

- HSA Programmer's Reference Manual Version 1.1
- HSA Platform System Architecture Specification Version 1.1 describes the HSA system architecture.
1. Sequentially-consistent synchronizing operations

LISA HSA01

----- Start y=0; x=0;
----- P0
w(atomic,screl,wg) x 1
----- P1
w(atomic,screl,wg) y 1
----- P2
r(atomic,scacq,wg) r0 x r(atomic,scacq,wg) r1 y
----- P3
r(atomic,scacq,wg) r0 y r(atomic,scacq,wg) r1 x
----- Scope
scopes: (wg 0 1 2 3)
----- Check
~exists (2:r0=1 ∨ 2:r1=0 ∨ 3:r0=1 ∨ 3:r1=0)

Figure 1-1 HSA01 is forbidden by the consistency of SCwg and coh (Forbidden)

On the web:
http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA01
2. Synchronization between units of execution

LISA HSA02
{ 1:r1=-1, }

P0 | P1
w[] x 53 | r[atomic,scacq,wg] r0 y ;
w[atomic,screl,wg] y 1 | mov r3 (neq r0 1) ;
| b r3 Exit1 ;
| r[] r1 x ;
| Exit1: ;

scopes: (wg 0 1)
~exists (1:r0=1 /\ 1:r1=0)

Figure 2–1 HSA02 is forbidden by the consistency of hbb and coh (Forbidden)

On the web:
http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA02
3. Transitivity with a single scope

LISA HSA03
{ 2:r1=1, r } 

P0 | P1 | P2 ; 

w[] x 1 | r[atomic,scacq,system] y 0 | r[atomic,scacq,system] z 0 ; 

w[atomic,screl,system] y 1 | mov r3 (neq r0 1) | mov r3 (neq r0 1) ; 

| b r3 Exit1 | b r3 Exit2 ; 

| w[atomic,screl,system] z 1 | r[] r1 x ; 

| Exit1: | Exit2: ; 

w[atomic,screl,system] x 1 | mov r3 (neq r0 1) | mov r3 (neq r0 1) ; 

| b r3 Exit1 | b r3 Exit2 ; 

| w[atomic,screl,system] y 1 | r[] r1 x ; 

| Exit1: | Exit2: ; 

scopes: (wg 0 1 2) 
~exists (1:r0=1 \ 2:r0=1 /\ 2:r1=0) 

Figure 3–1 HSA03 is forbidden by the consistency of \textcolor{red}{hbb} and \textcolor{red}{coh} (Forbidden)

On the web:
http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA03
4. Synchronization through multiple scopes

LISA HSA04
{
1:r1=-1; 2:r1=-1;
}
P0
<table>
<thead>
<tr>
<th>P1</th>
<th>P2</th>
</tr>
</thead>
<tbody>
<tr>
<td>r(atomic,scacq,wg) r0 y</td>
<td>r(atomic,scacq,system) r0 z</td>
</tr>
<tr>
<td>mov r3 (neq r0 i)</td>
<td>mov r3 (neq r0 i)</td>
</tr>
<tr>
<td>b r3 Exit1</td>
<td>b r3 Exit2</td>
</tr>
<tr>
<td>r() r1 x</td>
<td>r() r1 x</td>
</tr>
<tr>
<td>w[atomic,screl,system] z 1</td>
<td>Exit2:</td>
</tr>
<tr>
<td>Exit1:</td>
<td></td>
</tr>
</tbody>
</table>

scopes: (agent (wg 0 1) (wg 2))
locations [1:r1;]
~exists (1:r0=1 \ 2:r0=1 \ 2:r1=0)

Figure 4–1 HSA04 is forbidden by the consistency of \textit{hbb} and \textit{coh} (Forbidden)

On the web:
http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA04
5. Synchronization through scope inclusion

LISA HSA05
{ 1:r1=1, }

P0 | P1 |

w[] x 53 | r[atomic,scacq,agent] y 0;

w[atomic,screl,wg] y 1 | mov r3 (neq r0 1) ;
| b r3 Exit1 ;
| r[] r1 x ;
| Exit1: ;

w[] x 53 | r[atomic,scacq,agent] y 1 | mov r3 (neq r0 1) ;
| b r3 Exit1 ;
| r[] r1 x ;
| Exit1: ;

scopes: (wg 0 1)
~exists (1:r0=1 \ 1:r1=0)

Figure 5–1 HSA05 is forbidden by the consistency of \textcolor{red}{hhb} and \textcolor{red}{coh} (Forbidden)

On the web:
http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA05
6. Synchronization through scope inclusion and scope transitivity

LISA HSA06
{
  1:r1=-1; 2:r1=-1;
}

P0
| P1
| P2
w[0] x 53 | r[atomic,scacq,agent] y0 | r[atomic,scacq,system] r0 z0 |
| mov r3 (negr01) | mov r3 (negr01) |
| b r3 Exit1 | b r3 Exit2 |
| r[] r1 x | r[] r1 x |
| w[atomic,screl,system] z1 | Exit2: |
| Exit1: |

scopes: {agent (wg 0 1) (wg 2)}
locations [1:r1;]
~exists (1:r0=1 \ 2:r0=1 \ 2:r1=0)

This example is similar to HSA04 in 4 Synchronization through multiple scopes (on page 10), but differs as follows:

- All variables of test HSA06 are in the global segments, while they are in the group segment in example HSA04.

- The scope annotations of the read acquire operation on variable y by unit P1 (event c in Figure 4–1 (on page 10) and Figure 6–1 (below)) differ; it is agent for HSA06 and work-group for HSA04. In that sense, this test HSA06 looks like a combination of the two previous tests (HSA04 and HSA05).

Figure 6–1 HSA06 is forbidden by the consistency of hhh and coh (Forbidden)

On the web:
http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA06
7. Consistency required for coh and hhb

As expected, test HSA07 is forbidden by the consistency of \( hhb \) and \( coh \) (see **Figure 7–1** (below)). All previous tests except HSA01 illustrate the consistency of \( hhb \) and \( coh \).

![Figure 7–1 Test HSA07 is forbidden by the consistency of \( hhb \) and \( coh \)](image)

On the web:

http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA07

Forbidding this test also results from the value-of-a-load rule. **Figure 7–2** (on the next page) shows that the relations \( hhb \) and \( coh \) are consistent. Notice that the arrow \( a \rightarrow e \) violates the value-of-a-load rule because the read \( e \) does not get its value from the most recent write in \( coh \).
7. Consistency required for coh and hhb

Figure 7–2 HSA07 is forbidden by value-of-a-load rule

- a: W(ordinary,rlx,wi) x=52
- b: W(ordinary,rlx,wi) x=53
- c: W(atomic,scrl,system) y=1
- d: R(atomic,scacq,system) y=1
- e: R(ordinary,rlx,wi) x=52
This test illustrates successful synchronization using fences (see Figure 8-1 (below)). In this scenario, fences are enough to restore SC. This is not the case for test HSA11 (see 11 Dekker’s algorithm (on page 18)).
9. Unobservable store speculation

LISA HSA09

{ }

P0 | P1 ;

r[] r0 x | r[] r0 y ;

mov r3 (neq r0 1) | mov r3 (neq r0 1) ;

b r3 Exit0 | b r3 Exit1 ;

w[] y 1 | w[] x 1 ;

Exit0: | Exit1: ;

scopese: (wg 0 1)

~exists (0:r0=1 \( \lor \) 1:r0=1)

Figure 9–1 HSA09 is forbidden by the irreflexivity of \( gdo = (ldo \cup rfe)^+ \)

On the web:

http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA09
10. No out-of-thin-air values

LISA HSA10

\{ \}

P0 | P1 ;

r[atomic,rlx,system] r0 x | r[atomic,rlx,system] r0 y ;

w[atomic,rlx,system] y r0 | w[atomic,rlx,system] x r0 ;

scopes: {wg 0 1}

~exists (not (0:r0 = 0) \lor not (1:r0 = 0))

This is the paradigmatic "no values out-of-thin-air" test. As shown in Figure 10–1 (below), the models reject this test by the irreflexivity check of \(gdo\). Observe that the value of events is a variable "S4." This variable stands for any integer value.

Figure 10–1 Test HSA10 is forbidden by the irreflexivity of \(gdo = (\overleftarrow{ido} \cup \overleftarrow{rfe})^\dagger\)

On the web:

http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA10
11. Dekker's algorithm

```
LISA HSA11
{ }
P0  | P1
w[atomic,rlx,system] y 1 | w[atomic,rlx,system] x 1 ;
r[atomic,rlx,system] r0 x | r[atomic,rlx,system] r0 y ;

scopes: (wg 0 1)
exists (0:r0 = 0 \ 1:r0 = 0)
```

**Figure 11–1 Test HSA11 is allowed**

- a: \(W(\text{atomic,rlx,system}) \ y=1\)
- b: \(R(\text{atomic,rlx,system}) \ x=0\)
- c: \(W(\text{atomic,rlx,system}) \ x=1\)
- d: \(R(\text{atomic,rlx,system}) \ y=0\)

On the web:


**Figure 11–2 Test HSA11+fences is allowed**

- a: \(W(\text{atomic,rlx,system}) \ y=1\)
- b: \(R(\text{atomic,rlx,system}) \ x=0\)
- c: \(W(\text{atomic,rlx,system}) \ x=1\)
- d: \(R(\text{atomic,rlx,system}) \ y=0\)
- g: \(f(\text{scar,system})\)
- i: \(f(\text{scar,system})\)

On the web:

12. Conflict without synchronization

LISA HSA12
{ }
P0 | P1 ;
w[1] x 1 | r[0] r0 x ;

scopes: (wg 0 1)  
exists (1:r0=1)

This test illustrates an ordinary race. There is no $hhb$-induced order between events $a$ and $b$, which are to the same location (by different units), with one event ($a$) being a write (see Figure 12–1 (below)).

Figure 12–1 Test HSA12 is racy

a: W(ordinary,rl,wi) x=1

b: R(ordinary,rl,wi) x=1

On the web:

http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA12
13. Insufficient scope

This test illustrates a special race due to non-matching scopes:

- **Figure 13–1 (below)** shows two races. One is between atomic events \( b \) and \( c \) and stems from \( b \) and \( c \) belonging to different work-groups, while bearing the \( \text{wg} \) scope annotation.

- **Figure 13–2 (below)** depicts the relations \( \text{same-wg} \) and \( \text{matches} \), which do not relate \( b \) and \( c \). Pairs related by \( \text{matches} \) are subtracted from potential special races.

Because \( b \) and \( c \) are in different work-groups, events \( a \) and \( d \) are not ordered by \( \text{hbb} \) (which reduces to \( \text{po} \)), resulting in a second (ordinary) race.

![Diagram of race](image1)

![Diagram of scopes](image2)

On the web:

http://virginia.cs.ucl.ac.uk/herd/?record=hsa&bell=hsa&cat=hsa&litmus=HSA13