Multics Change Request

Subject: MCR10050, Increase loop_lock_time size.

Author: Charles Anthony

Date: May 31, 2018

Introduction

The “lockless” multi-threaded emulator experiences a overflow fault when updating tc_data$loop_lock_time in pxss alm after a week heavy load in a 4 CPU configuration.

It is thought that the lock time may increasing atypically fast due to host CPU starvation; a CPU loops on lock acquisition while the CPU holding the lock is blocked; a condition that would not arise in the canonical hardware implementation.

Problem

The variable tc_data$loop_lock_time tracks the amount of time that pxss spends trying to acquire the APT lock. The code that accumulates the time is in pxss$READ_LOCK and pxss$WRITE_LOCK. Both routines start with the code:

```
read_clock " setup for metering looplocks
staq    temp
```

Both routines share exit code containing:

```
read_clock
sbaq    temp
asq     tc_data$loop_lock_time
```

tc_data$loop_lock_time is declared in tcm.incl.pl1:

```
2 loop_lock_time fixed bin (18), /* time looping on the APT lock */
```

The ASQ instruction will perform a thirty-six-bit bit signed add (despite the “(18)” declaration), and will overflow fault when the accumulated time exceeds thirty-five bits.
Proposed Changes

Redefine `tc_data$loop_lock_time` to seventy one bits. This is a straight-forward change, already implemented in several related time accumulation variables.

```
2 loop_lock_time fixed bin (71), /* time looping on the APT lock */
```

The code that accumulates time is changed to do double-word arithmetic.

```
read_clock
sbaq temp
adaq tc_data$loop_lock_time
staq tc_data$loop_lock_time
```

The metering code uses various accumulated times in “delta” calculations, subtracting the values sampled at two points in time, storing the result in a “float bin”, typically:

```
    time = tcdp2 -> tcm.loop_lock_time - tcdpl -> tcm.loop_lock_time;
```

By redefining `loop_lock_time` as “fixed bin (71)”, the correct code will be generated so no changes are needed in the metering code to accommodate this change.

Because the APT lock has both read and write lock modes, it is possible for a race condition to occur when a CPU has acquired a read lock at the same time as a CPU has acquired a write lock, and the updating of `loop_lock_time` is done incorrectly due to the non-atomic nature of the ADAQ/STAQ instruction sequence.

If that race condition occurs, the accumulated time from one of the CPUs may be overwritten by the other, cause the loss of the metering information from one of the CPUs.

Since only a single sample would be lost per race condition, the loss small loss of data should be acceptable when compared to the complexity and overhead of 72 bit atomic arithmetic.

Detailed changes

```
cpa >ldd>include>tcm.incl.pl1 tcm.incl.pl1
```
Changed by B to:
B8    /* Modified 2018.05.31 by Charles Anthony to promote loop_lock_time to (71) */

A39    2 loop_lock_time fixed bin (18),                        /* time
looping on the APT lock */
Changed by B to:
B39    2 obs_loop_lock_time fixed bin (18),                    /* obsolete; promoted to (71) */

A242    2 pad5 (176) fixed bin (35),                            /* room
for expansion compatibly */
Changed by B to:
B242    2 loop_lock_time fixed bin (71),                        /* time
looping on the APT lock */
B243    2 pad5 (174) fixed bin (35),                            /* room
for expansion compatibly */

Comparison finished: 3 differences, 7 lines.

cpa >ldd>include>tc_meters.incl.alm tc_meters.incl.alm

A39                 equ       loop_lock_time,31
Changed by B to:
B39                 equ       obs_loop_lock_time,31

Inserted in B:
B227                equ       loop_lock_time,336            DOUBLE
Preceding:
A227
A228      "500 octal

Comparison finished: 2 differences, 3 lines.

cpa pxss.alm.orig pxss.alm

Inserted in B:
B106      "  5) change(2018-05-31,Anthony):Charles Anthony to promote
loop_lock_time to
B107      "     (71).
Preceding:
A106      "                                                      END HISTORY
COMMENTS

A1407               asq       tc_data$loop_lock_time
Changed by B to:
Comparison finished: 2 differences, 5 lines.

Bug Reference

Ticket #114 “tc_data$loop_lock_time overflows”

Version History

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision</th>
<th>Author</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-05-30</td>
<td>1.0</td>
<td>Charles Anthony</td>
<td>Initial version of MCR</td>
</tr>
<tr>
<td>2018-05-30</td>
<td>1.1</td>
<td>Charles Anthony</td>
<td>Fix typos in tcm.incl.pl1 and pxss.alm; add missing tc_meters.incl.alm.</td>
</tr>
<tr>
<td>2018-06-4</td>
<td>1.2</td>
<td>Charles Anthony</td>
<td>Document race condition.</td>
</tr>
</tbody>
</table>