

MARTe2-core - User story #123

MARTe-EX-F-1.1.8: A framework developer shall be capable of querying the family and model of the CPU where MARTe is running.

29.05.2015 14:19 - André Neto

Status: Closed	Start date:
Priority: Normal	Due date:
Assignee:	% Done: 0%
Category:	Estimated time: 2.00 hours
Target version: 0.2	Spent time: 0.00 hour
Git branch (link):	SVN commit (link/?p=rev):
Git merge to develop (link):	

Description

Implement requirement MARTe-EX-F-1.1.8 based on the original MARTe implementation

Note: Being able to know the number CPU family and model may allow framework developers to take run-time decisions about the distribution of resources (e.g. availability of operating system specific features).

NOTE: User story moved back to implementation in order to fix complex bug (found by Llorenc)

Source code files modified

- Source\Core\L0Portability\Architecture\x86_cl\ProcessorA.h
- Source\Core\L0Portability\Architecture\x86_gcc\ProcessorA.h
- Source\Core\L0Portability\OperatingSystem\Linux\ProcessorOS.h
- Source\Core\L0Portability\OperatingSystem\Windows\ProcessorOS.h
- Source\Core\L0Portability\Processor.cpp
- Source\Core\L0Portability\Processor.h
- Source\Core\L0Portability\ProcessorType.cpp
- Source\Core\L0Portability\ProcessorType.h
- Test\Core\L0Portability\ProcessorTest.h
- Test\Core\L0Portability\ProcessorTest.cpp
- Test\Core\L0Portability\ProcessorTypeTest.h
- Test\Core\L0Portability\ProcessorTypeTest.cpp
- Test\GTest\ProcessorGTest.cpp
- Test\GTest\ProcessorTypeGTest.cpp

Architecture & design review

Date of the review: 12/06/2015

Person who did the review: André Neto

Version of architecture & design document: N/A. As per the original MARTe implementation, this is a support function and does not require formal design in UML.

Result of review: N/A

Code and documentation review

Date of the review: 18/06/2015

Person who did the review: Ivan Herrero

Result of review: PASS

List of non-conformities: N/A

Unit test review

Date of the review: 01/07/2015

Person who did the review: André Neto

Result of coverage tests review: PASS [100%]

Result of functional tests review: PASS [100%]

Result of review: PASS

List of non-conformities: N/A

See [#122](#)

Associated revisions

Revision 58852254 - 15.06.2015 11:53 - Giuseppe Ferrò

Added processor related functions. #122 #123

Revision 142aa3bd - 17.06.2015 13:23 - Giuseppe Ferrò

Formatted code using new templates.
#122 #123

Revision 886334a7 - 17.06.2015 14:59 - Giuseppe Ferrò

Moved a global variable into the class.
#122 #123

Revision 3573b526 - 18.06.2015 10:24 - Giuseppe Ferrò

Fixed errors accordingly with revision.
#122 #123

Revision 56f896af - 23.06.2015 16:59 - Giuseppe Ferrò

Added Threads and ThreadsDatabase tests.
#110 #123 #103 #104 #112 #107 #108 #111 #106 #105 #109 #113

Revision a1ee0477 - 25.06.2015 13:37 - Giuseppe Ferrò

Added Threads unit tests.
Renamed previous tests because they are integration tests.
#110 #123 #103 #104 #112 #107 #108 #111 #106 #105 #109 #113

Revision 3d7d5f94 - 26.06.2015 11:58 - Giuseppe Ferrò

Added Processor and ProcessorType tests.
#122 #123

Revision 95c810f0 - 26.06.2015 11:59 - Giuseppe Ferrò

Added Processor and ProcessorType GTests.
#122 #123

Revision ba754568 - 15.07.2015 17:21 - Llorenc Capella

US #123 - Processor and ProcessorType test updated.

List of changes: * Printf added on the ProcessorTest.cpp in order check that the information about the CUP is correct. * Minor modification on the ProcessorTypeTest.cpp. * Format files.

Revision ce744402 - 26.07.2015 17:05 - André Neto

Fixed processor model reported in #123. Implementation of #156.

The bug related to the computation of the processor model was fixed (see comment in #123).

As part of the linting process (#156) the code structure was slightly changed in order to delegate the implementation of the Processor

operating system functionality to a .cpp unit. This improves the consistency with the other operating system classes.

History

#1 - 03.06.2015 11:32 - Riccardo Vitelli

- Target version changed from Backlog to 0.1
- Start date deleted (29.05.2015)
- Estimated time set to 2.00 h

#2 - 12.06.2015 10:31 - Riccardo Vitelli

- Status changed from New to Arch: Rev

#3 - 12.06.2015 11:15 - Riccardo Vitelli

- Assignee set to André Neto

#4 - 12.06.2015 18:27 - André Neto

- Description updated

#5 - 15.06.2015 10:10 - Riccardo Vitelli

- Status changed from Arch: Rev to Code: Impl

#6 - 15.06.2015 10:10 - Riccardo Vitelli

- Assignee changed from André Neto to Giuseppe Ferro

#7 - 17.06.2015 12:23 - Ivan Herrero

- Description updated

#8 - 17.06.2015 18:32 - Ivan Herrero

- Description updated

#9 - 18.06.2015 09:34 - Ivan Herrero

- Status changed from Code: Impl to Code: Rev

#10 - 18.06.2015 15:03 - Ivan Herrero

- Description updated

#11 - 18.06.2015 15:08 - Ivan Herrero

- Status changed from Code: Rev to Unit: Impl

#12 - 18.06.2015 15:09 - Ivan Herrero

- Assignee deleted (Giuseppe Ferro)

#13 - 18.06.2015 15:21 - Ivan Herrero

- Description updated

#14 - 18.06.2015 18:18 - Ivan Herrero

Code and documentation review passed. Unit testing implementation pending.

#15 - 25.06.2015 17:22 - Riccardo Vitelli

- Status changed from Unit: Impl to Unit: Rev

#16 - 01.07.2015 08:29 - André Neto

- Assignee set to André Neto

#17 - 01.07.2015 09:02 - André Neto

- Description updated

#18 - 01.07.2015 10:07 - Riccardo Vitelli

- Status changed from Unit: Rev to Unit: Impl

#19 - 01.07.2015 10:07 - Riccardo Vitelli

- Assignee deleted (André Neto)

#20 - 02.07.2015 18:06 - Riccardo Vitelli

- Target version changed from 0.1 to 0.2

#21 - 15.07.2015 17:24 - Llorenc Capella

- Assignee set to Llorenc Capella

#22 - 15.07.2015 17:54 - Llorenc Capella

- Status changed from Unit: Impl to Unit: Rev

#23 - 15.07.2015 18:01 - Llorenc Capella

- Assignee deleted (Llorenc Capella)

The ProcessorTest only check that the returned values of the class are meaningless, but does not check if the values are correct...
I print the returned values on the screen and I compared them with the lscpu. As a result I found that the Model from the test and using the lscpu command is different.

#24 - 16.07.2015 09:25 - Riccardo Vitelli

- Status changed from Unit: Rev to Code: Impl

#25 - 16.07.2015 09:27 - Riccardo Vitelli

- Description updated

#26 - 20.07.2015 11:30 - Ivan Herrero

Llorenc Capella wrote:

The ProcessorTest only check that the returned values of the class are meaningless, but does not check if the values are correct...
I print the returned values on the screen and I compared them with the lscpu. As a result I found that the Model from the test and using the lscpu command is different.

In fact there is a bug in the implementation, or better, the implementation is not complete:

Intel and AMD have suggested applications to display the family of a CPU as the sum of the "Family" and the "Extended Family" fields shown above, and the model as the sum of the "Model" and the 4-bit left-shifted "Extended Model" fields. If "Family" is different than 6 or 15, only the "Family" and "Model" fields should be used while the "Extended Family" and "Extended Model" bits are reserved. If "Family" is set to 15, then "Extended Family" and the 4-bit left-shifted "Extended Model" should be added to the respective base values, and if "Family" is set to 6, then only the 4-bit left-shifted "Extended Model" should be added to "Model". [Source: <https://en.wikipedia.org/wiki/CPUID>]

It needs to be reimplemented.

#27 - 29.07.2015 15:12 - Riccardo Vitelli

- Description updated

#28 - 29.07.2015 15:12 - Riccardo Vitelli

- Status changed from Code: Impl to Closed