

Overview

ESESC Tutorial

Speaker: Jose Renau



*Department of Computer Engineering,
University of California, Santa Cruz*
<http://masc.soe.ucsc.edu>



Tutorial Logistics

08:00 - 08:30: Breakfast

08:30 - 09:00: Overview

09:00 - 09:30: Building and Running

09:30 - 10:00: Validation

10:00 - 10:30: Morning Break

10:30 - 11:30: Memory Hierarchy and Cache Coherence

11:30 - 12:00: Power

12:00 - 13:00: Lunch

13:00 - 14:00: GPU

14:00 - 15:00: Sampling

15:00 - 15:30: Afternoon Break

15:30 - 16:00: Thermal Model

16:00 - 17:00: Recent Projects and Wrap-up

ESESC Logistics

- ESESC blog has these slides

<http://masc.soe.ucsc.edu/esesc>

- ESESC forum

<https://groups.google.com/forum/#!forum/esesc>

- ESESC repository at github

<https://github.com/masc-ucsc/esesc>

- To get the code

```
git clone https://github.com/masc-ucsc/esesc.git
```

What is ESESC?

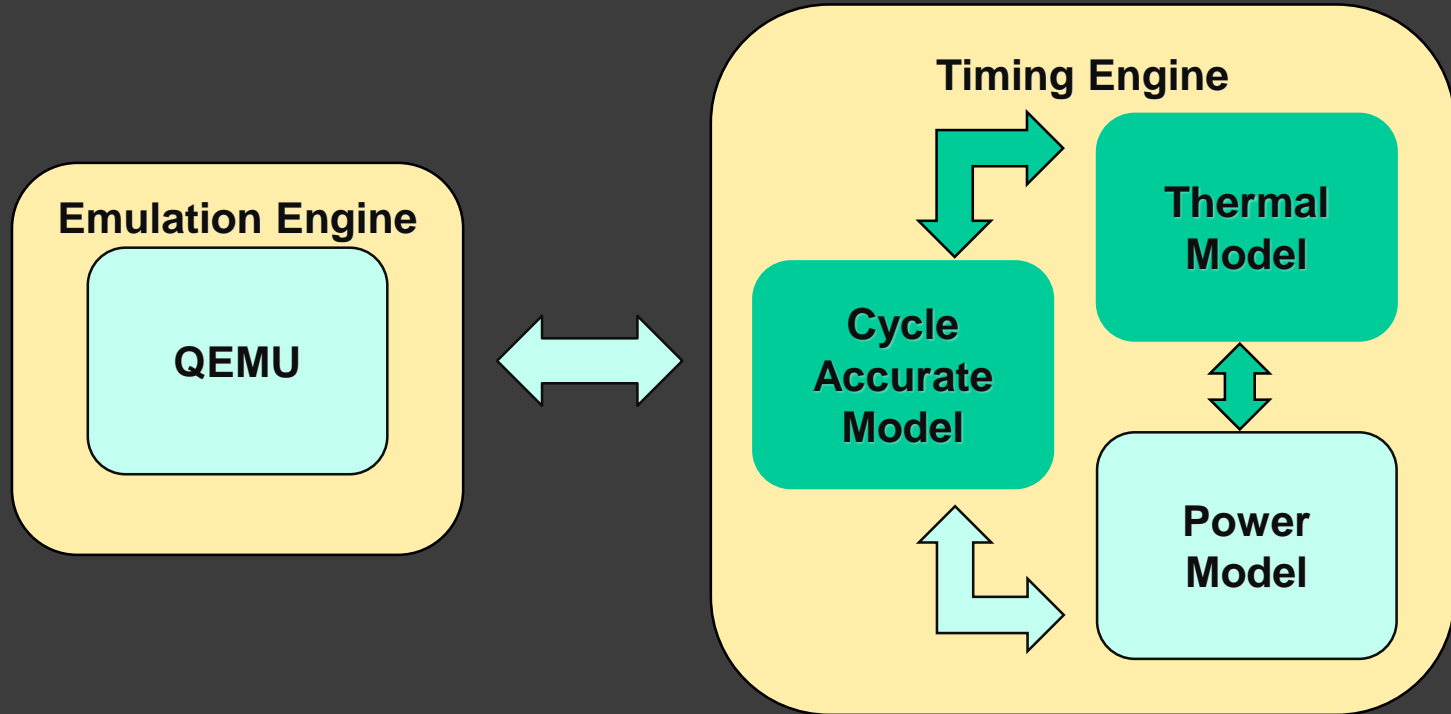
- FAST cycle accurate chip multiprocessor
- In-order and out-of-order processors
- GPUs (beta)
- Performance/Power/Thermal models

ESESC = Enhanced SESC

- Same goals as SESC
 - Fast cycle-accurate simulator
 - Easy to understand and extend
 - Multiple configurations available

- Many enhancements...

Execution-Driven Simulation



ESESC Enhancements

- Runs unmodified ARM binaries
- Statistical sampling
- New memory hierarchy design
- Integrated thermal model
- McPAT power model
- Heterogeneous models
 - In-order, Out-of-order, GPU

Unmodified ARM binaries

- SESC
 - Custom MIPS-based compilation flow
- ESESC
 - Unmodified ARM Linux binaries
 - Cracks ARM instructions to ESESC uOPs

Executable Compilation Platform



Fast Simulation

- ESESC achieves over 50 MIPS
 - Significant effort creating an efficient timing
- Many sampling techniques available

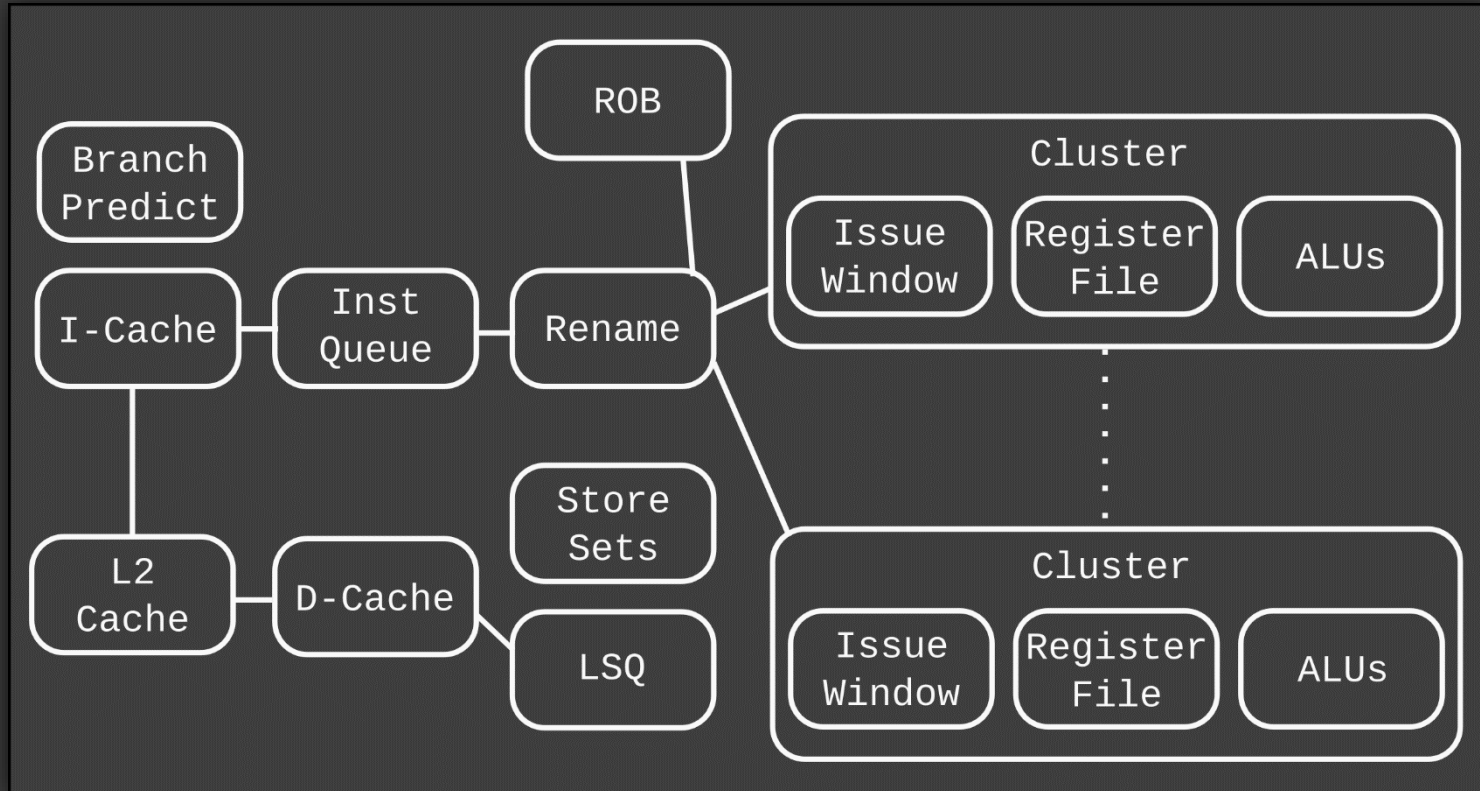
Code Structure

- `<esesc root>/docs`
- `<esesc root>/conf`
 - Configuration files
- `<esesc root>/emul`
 - Source code and libraries for the emulator
- `<esesc root>/simu`
 - Source for timing model
- `<esesc root>/main`
 - Top level code directory
- ...

New Compilation Setup for ESESC

- Runs in Linux x86-64 and ARMv7
 - Not tested in OS X and Linux x86-32
- Build using CMAKE
- Debug build
 - Slower
 - debug information
 - assertions / warnings
- Release build
 - ~5 times faster than debug
 - No assertions

Enhanced Out-of-order Core



Same Configuration File (esesc.conf)

```
[tradCORE]
# not showing all parameters
robSize = 128
cluster[0]      = 'Aunit'
...
cluster[N]      = 'Nunit'
...

[Aunit]
...
winSize         = 16+4
nRegs           = 128
iAALULat       = 1
iAALUUnit       = 'AUNIT_AALU'

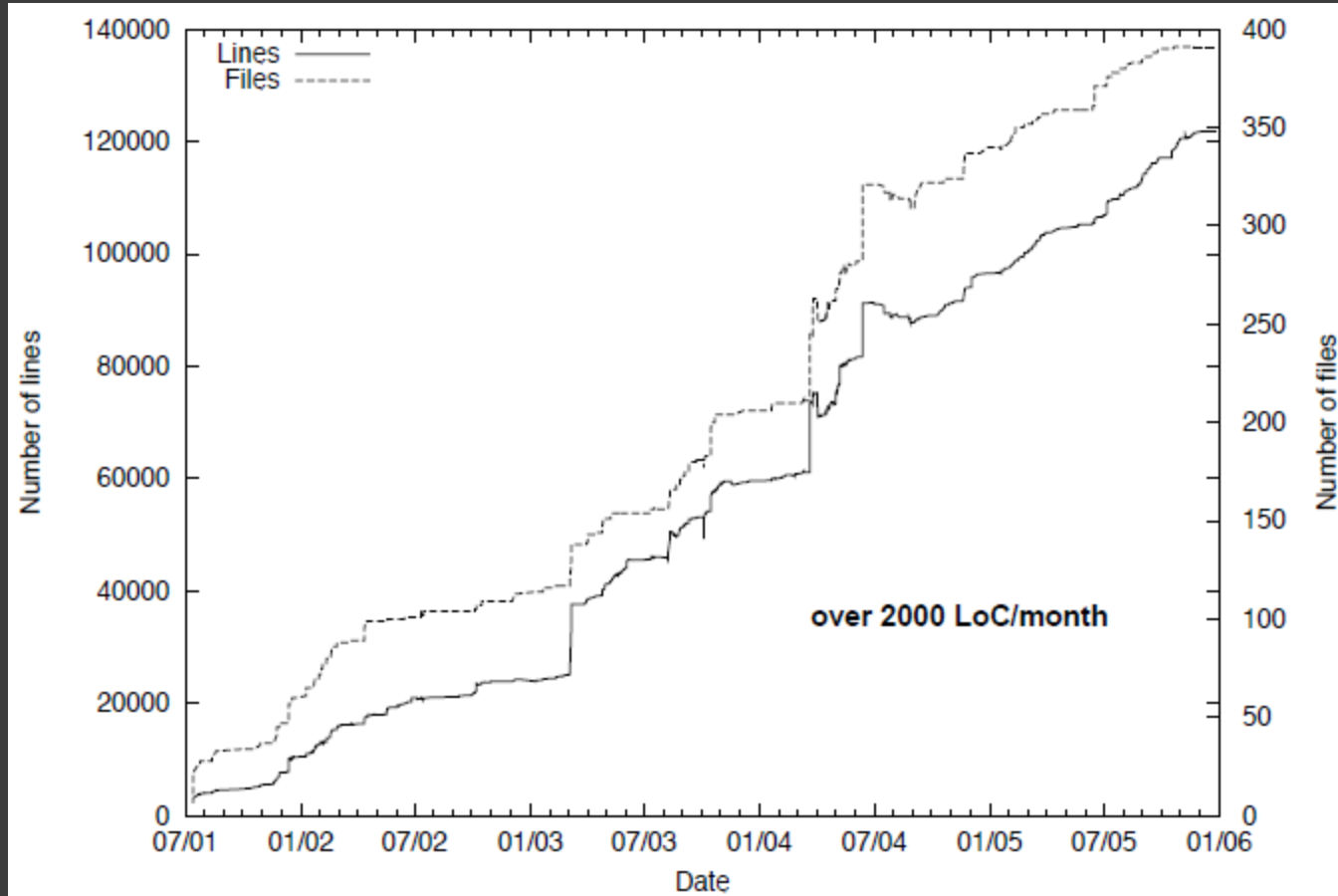
[AUNIT_AALU]
Num             = 1
Occ             = 1
```



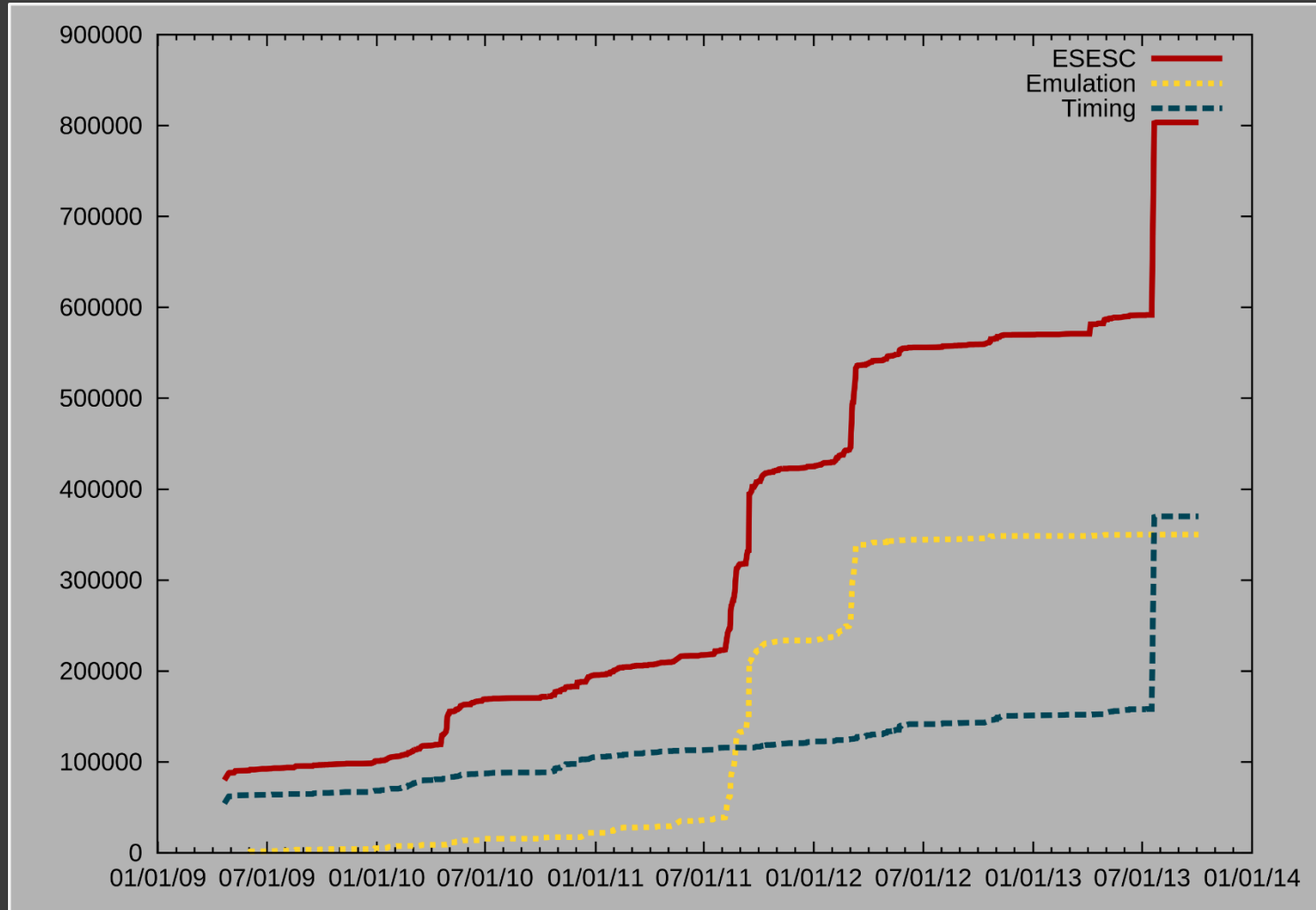
ESESC vs other Cycle-Accurate Sims

Simulator	Models	ISA	Full-System	Key-Feature	Speed
ESESC	Perf/Pwr/Temp	ARMv7	No (it was)	Sampling	~50MIPS
SESC	Per/Pwr/Temp	MIPS	No	Fast	~1MIP
gem5	Perf/Pwr	X86/ARM	Yes		
MARSSx86	Perf	X86	Yes		~200KIPS
Flexus	Perf?/Pwr?	SPARC	Yes	Sampling	
Multi2sim	Perf	X86	No	Heterogenous	
Sniper	Perf/Pwr	X86	No (pin)	Multithreaded	~10MIPS

SESC was a large project



ESESC has over 800KLoC Changes

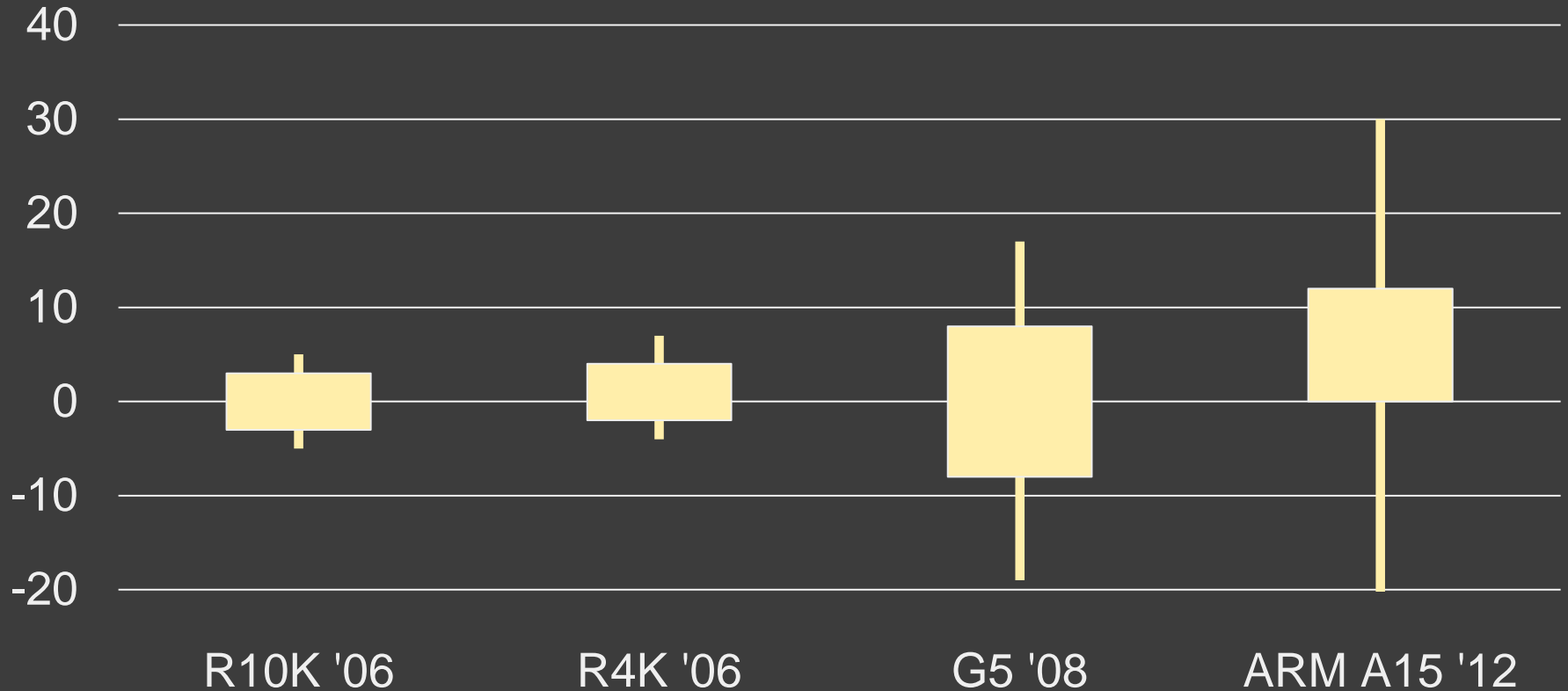


ESESC Usage Sample

```
Aggr.uIPC ProgressedTime globalClock
esesc_testing.U0vLxz AggrR 2.98 0.000000e+00 4.453447e+06
-----
#####
Cache          Occ AvgMemLat MemAccesses MissRate ( RD, WR, BUS) %%DMemAccMB/s
IL1(0)         0.0 2.9      4146594      6.29% ( 93.7%, 0.0%, 0.0%) 101.50%
-----
DL1(0)         0.0 5.5      3706526      0.36% ( 99.6%, 99.8%, 0.0%) 90.84%
-----
L2(0)         0.0 14.8     278147       10.47% ( 89.5%, 89.8%, 0.0%) 6.73%
L3            0.0 46.4     37691        4.56% ( 95.3%, 58.3%, 0.0%) 0.76%
-----
mada3:~/build/release/run$
mada3:~/build/release/run$rm esesc_*
mada3:~/build/release/run$ll
total 2.0M
-rw-r--r-- 1 renau faculty 14K Sep 27 10:12 cpu2000spMaxk100.conf
-rw-r--r-- 1 renau faculty 8.7K Sep 27 10:12 cpu2000spMaxk10.conf
-rw-r--r-- 1 renau faculty 9.6K Sep 27 10:12 cpu2000spMaxk30.conf
-rw-r--r-- 1 renau faculty 6.6K Sep 27 10:12 cpu2006sp.conf
-rwxr-xr-x 1 renau faculty 780K Sep 27 10:12 crafty.armel
-rw-r--r-- 1 renau faculty 413 Sep 27 10:12 crafty.in
-rwxr-xr-x 1 renau faculty 757K Sep 27 10:12 crafty_sparc32
-rw-r--r-- 1 renau faculty 4.4K Sep 27 10:12 esesc.conf
-rw-r--r-- 1 renau faculty 236K Sep 27 10:12 flp.conf
-rw-r--r-- 1 renau faculty 0 Sep 27 10:13 game.001
-rw-r--r-- 1 renau faculty 3.8K Sep 27 10:12 gpu.conf
-rw-r--r-- 1 renau faculty 847 Oct 4 10:39 memory-arch.dot
-rw-r--r-- 1 renau faculty 1.4K Sep 27 10:12 peq1.conf
-rw-r--r-- 1 renau faculty 24K Sep 27 10:12 shared.conf
-rw-r--r-- 1 renau faculty 18K Sep 27 10:12 therm.conf
-rw-r--r-- 1 renau faculty 26K Sep 27 10:12 therm_lp.conf
-rw-r--r-- 1 renau faculty 14K Sep 27 10:12 tsample.conf
-rw-r--r-- 1 renau faculty 13K Sep 27 10:12 tsample_lp.conf
mada3:~/build/release/run$../main/esesc <^C
mada3:~/build/release/run$ll
total 2.0M
-rw-r--r-- 1 renau faculty 14K Sep 27 10:12 cpu2000spMaxk100.conf
-rw-r--r-- 1 renau faculty 8.7K Sep 27 10:12 cpu2000spMaxk10.conf
-rw-r--r-- 1 renau faculty 9.6K Sep 27 10:12 cpu2000spMaxk30.conf
-rw-r--r-- 1 renau faculty 6.6K Sep 27 10:12 cpu2006sp.conf
-rwxr-xr-x 1 renau faculty 780K Sep 27 10:12 crafty.armel
-rw-r--r-- 1 renau faculty 413 Sep 27 10:12 crafty.in
-rwxr-xr-x 1 renau faculty 757K Sep 27 10:12 crafty_sparc32
-rw-r--r-- 1 renau faculty 4.4K Sep 27 10:12 esesc.conf
-rw-r--r-- 1 renau faculty 236K Sep 27 10:12 flp.conf
-rw-r--r-- 1 renau faculty 0 Sep 27 10:13 game.001
-rw-r--r-- 1 renau faculty 3.8K Sep 27 10:12 gpu.conf
-rw-r--r-- 1 renau faculty 847 Oct 4 10:39 memory-arch.dot
-rw-r--r-- 1 renau faculty 1.4K Sep 27 10:12 peq1.conf
-rw-r--r-- 1 renau faculty 24K Sep 27 10:12 shared.conf
-rw-r--r-- 1 renau faculty 18K Sep 27 10:12 therm.conf
-rw-r--r-- 1 renau faculty 26K Sep 27 10:12 therm_lp.conf
-rw-r--r-- 1 renau faculty 14K Sep 27 10:12 tsample.conf
-rw-r--r-- 1 renau faculty 13K Sep 27 10:12 tsample_lp.conf
mada3:~/build/release/run$../main/esesc <crafty.in
```

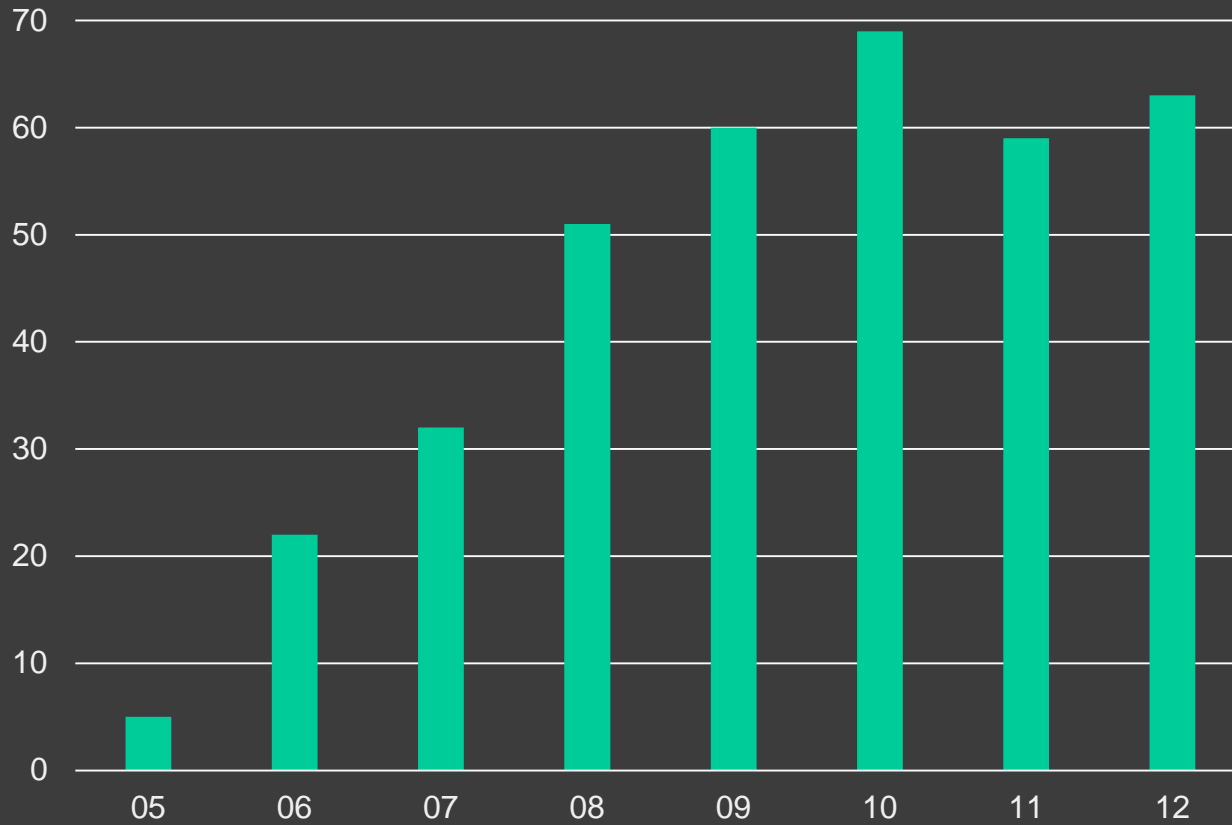
Validation (More Later)

% error total execution time

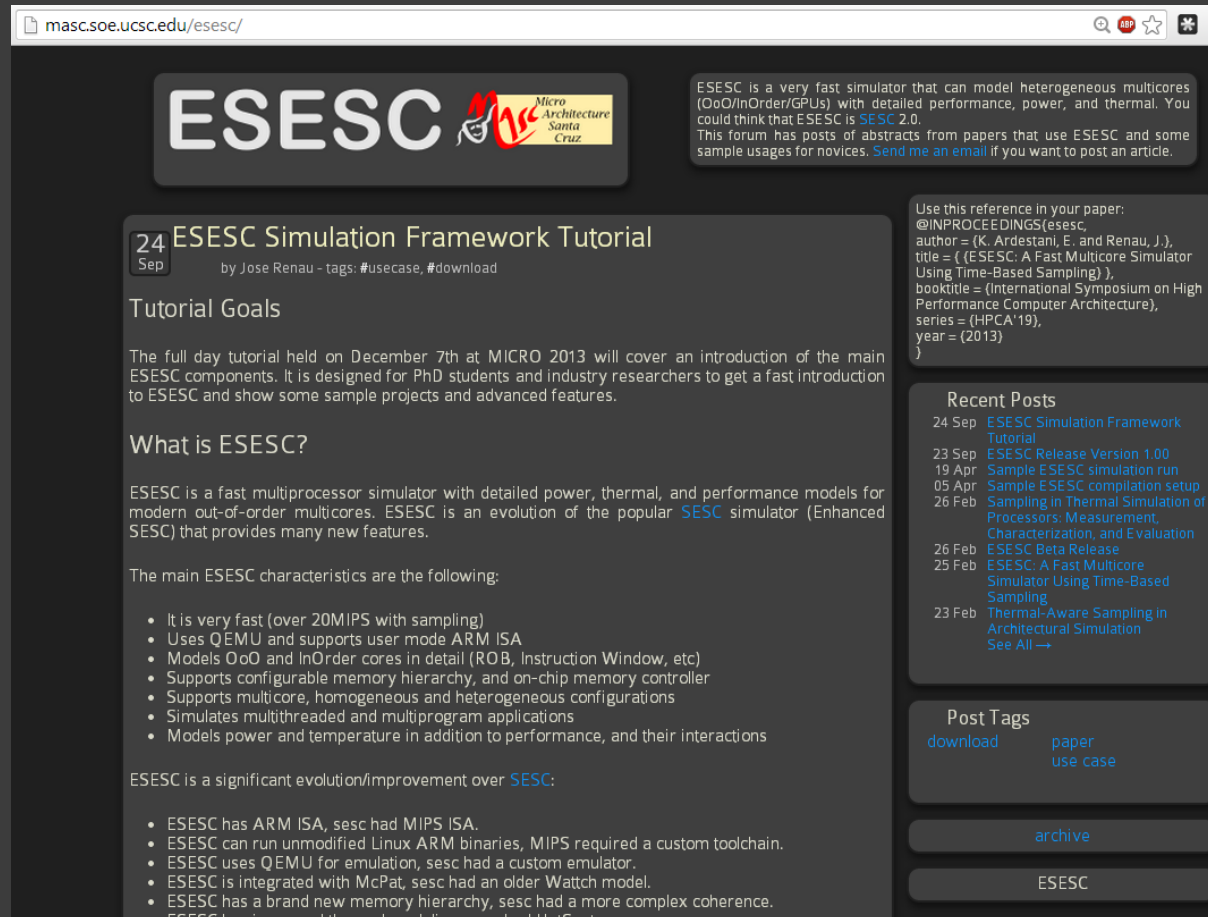


None of these class projects required to change a line of code.
Just configuration parameters

Papers using SESC



<http://masc.soe.ucsc.edu/esesc>



The screenshot shows the ESESC blog homepage. At the top left is the ESESC logo with the Micro Architecture Santa Cruz logo. To the right is a text box describing ESESC as a fast simulator for heterogeneous multicores. The main content area features a post titled 'ESESC Simulation Framework Tutorial' by Jose Renau, dated 24 Sep. Below the title are sections for 'Tutorial Goals', 'What is ESESC?', and a list of characteristics. A 'Recent Posts' sidebar on the right lists several articles. At the bottom, there are 'Post Tags' for 'download', 'paper', and 'use case', and a 'Recent Posts' section with a 'See All' link.

masc.soe.ucsc.edu/esesc/

ESESC

Micro Architecture Santa Cruz

ESESC is a very fast simulator that can model heterogeneous multicores (OoO/InOrder/GPUs) with detailed performance, power, and thermal. You could think that ESESC is [SESC 2.0](#). This forum has posts of abstracts from papers that use ESESC and some sample usages for novices. [Send me an email](#) if you want to post an article.

24 Sep ESESC Simulation Framework Tutorial

by Jose Renau - tags: [#usecase](#), [#download](#)

Tutorial Goals

The full day tutorial held on December 7th at MICRO 2013 will cover an introduction of the main ESESC components. It is designed for PhD students and industry researchers to get a fast introduction to ESESC and show some sample projects and advanced features.

What is ESESC?

ESESC is a fast multiprocessor simulator with detailed power, thermal, and performance models for modern out-of-order multicores. ESESC is an evolution of the popular [SESC](#) simulator (Enhanced SESC) that provides many new features.

The main ESESC characteristics are the following:

- It is very fast (over 20MIPS with sampling)
- Uses QEMU and supports user mode ARM ISA
- Models OoO and InOrder cores in detail (ROB, Instruction Window, etc)
- Supports configurable memory hierarchy, and on-chip memory controller
- Supports multicore, homogeneous and heterogeneous configurations
- Simulates multithreaded and multiprogram applications
- Models power and temperature in addition to performance, and their interactions

ESESC is a significant evolution/improvement over [SESC](#):

- ESESC has ARM ISA, sesc had MIPS ISA.
- ESESC can run unmodified Linux ARM binaries, MIPS required a custom toolchain.
- ESESC uses QEMU for emulation, sesc had a custom emulator.
- ESESC is integrated with McPat, sesc had an older Wattech model.
- ESESC has a brand new memory hierarchy, sesc had a more complex coherence.
- ESESC has improved thermal modeling, sesc had HotSpot

Use this reference in your paper:

```
@INPROCEEDINGS(esesc,
author = (K. Ardestani, E. and Renau, J.),
title = {(ESESC: A Fast Multicore Simulator
Using Time-Based Sampling)},
booktitle = (International Symposium on High
Performance Computer Architecture),
series = (HPCA'19),
year = (2013)
}
```

Recent Posts

- 24 Sep [ESESC Simulation Framework Tutorial](#)
- 23 Sep [ESESC Release Version 1.00](#)
- 19 Apr [Sample ESESC simulation run](#)
- 08 Apr [Sample ESESC compilation setup](#)
- 26 Feb [Sampling in Thermal Simulation of Processors: Measurement, Characterization, and Evaluation](#)
- 26 Feb [ESESC Beta Release](#)
- 25 Feb [ESESC: A Fast Multicore Simulator Using Time-Based Sampling](#)
- 23 Feb [Thermal-Aware Sampling in Architectural Simulation](#)

[See All →](#)

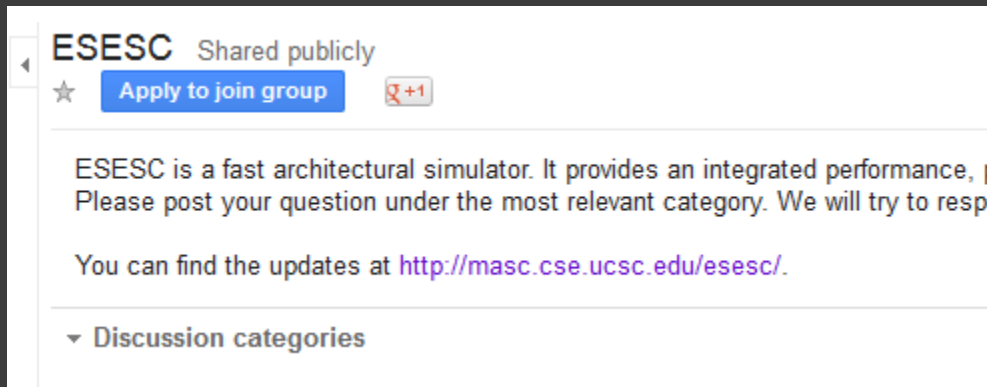
Post Tags


- [download](#)
- [paper](#)
- [use case](#)

Recent Posts

- [archive](#)
- [ESESC](#)

<https://groups.google.com/forum/#!forum/esesc>

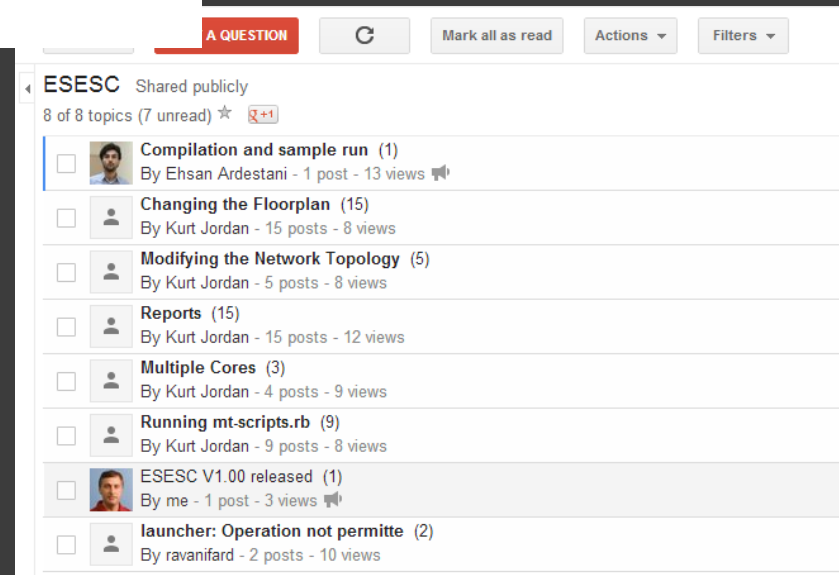



ESESC Shared publicly
★ [Apply to join group](#) 


ESESC is a fast architectural simulator. It provides an integrated performance, power, and area analysis. Please post your question under the most relevant category. We will try to respond as quickly as possible.









You can find the updates at <http://masc.cse.ucsc.edu/esesc/>.

▼ Discussion categories



A QUESTION  [Mark all as read](#) [Actions](#) [Filters](#)

ESESC Shared publicly
8 of 8 topics (7 unread) ★ 

-  **Compilation and sample run** (1)
By Ehsan Ardestani - 1 post - 13 views
-  **Changing the Floorplan** (15)
By Kurt Jordan - 15 posts - 8 views
-  **Modifying the Network Topology** (5)
By Kurt Jordan - 5 posts - 8 views
-  **Reports** (15)
By Kurt Jordan - 15 posts - 12 views
-  **Multiple Cores** (3)
By Kurt Jordan - 4 posts - 9 views
-  **Running mt-scripts.rb** (9)
By Kurt Jordan - 9 posts - 8 views
-  **ESESC V1.00 released** (1)
By me - 1 post - 3 views
-  **launcher: Operation not permitte** (2)
By ravanifard - 2 posts - 10 views

ESESC Public Repository

<https://github.com/masc-ucsc/esesc>

masc-ucsc / esesc

ESESC: A Fast Multicore Simulator <http://masc.soe.ucsc.edu/esesc/> — Edit

33 commits 1 branch 0 releases 1 contributor

branch: master esesc / +

Commit Message	Author	Time
update to README	southerngs	7 days ago
bins	More code cleanup in preparation for MICRO demo	8 days ago
conf	Converting documentation to Markdown	7 days ago
docs	Converting documentation to Markdown	7 days ago
emul	More code cleanup in preparation for MICRO demo	8 days ago
gold	More code cleanup in preparation for MICRO demo	8 days ago
main	More code cleanup in preparation for MICRO demo	8 days ago
misc	More code cleanup in preparation for MICRO demo	8 days ago

Remember to cite

- If you use ESESC, cite this paper:
- [ESESC: A Fast Multicore Simulator Using Time-Based Sampling](#), Ehsan K.Ardestani, and Jose Renau, International Symposium on High-Performance Computer Architecture (**HPCA**), February 2013.

Outline

08:00 - 08:30: Breakfast

~~08:30 - 09:00: Overview~~

09:00 - 09:30: Building and Running

09:30 - 10:00: Validation

10:00 - 10:30: Morning Break

10:30 - 11:30: Memory Hierarchy and Cache Coherence

11:30 - 12:00: Power

12:00 - 13:00: Lunch

13:00 - 14:00: GPU

14:00 - 15:00: Sampling

15:00 - 15:30: Afternoon Break

15:30 - 16:00: Thermal Model

16:00 - 17:00: Recent Projects and Wrap-up

Demo Accounts for the Tutorial

- Get a demo/password
- Shared server, be nice submitting jobs

```
ssh demo???@soe.ucsc.edu
```

Questions

- Remember to ask questions during talks