Linaro Connect Q4'12

Peeking into your Linux System

Vincent Guittot <vincent.guittot@linaro.org>
Linaro Power Management Working Group

Tools

- Used during this sessions
  - ftrace / trace-cmd
  - kernelshark
  - cylictest
  - sysbench
  - taskset
  - IRQ affinity
  - arm-probe

Hardware

- TC2 board
  - Access to big and LITTLE power domain
- ARM probe
  - Up to 3 channels
  - 10kHz sampling rate

Software

- Full Ubuntu Image 12.09
- Linaro ARM Landing Team kernel
- Disable HMP task placement
  - Disable load balance between cluster
- Additional patch set

Goals

- Spy system behavior
- Understand wake up sources
- Understand scheduling behavior
- Exercise your system

ftrace overview

- ftrace
  - In kernel tracing utility
  - Trace specific events like entering an idle state..
  - Trace function call
  - And more ...

- Location
  - <debugfs path>/tracing/

ftrace overview

- Useful articles in addition to Documentation:
  - http://lwn.net/Articles/366796/
  - http://lwn.net/Articles/365835/
- Use trace-cmd tool
  - Simplify ftrace configuration and use

trace-cmd overview

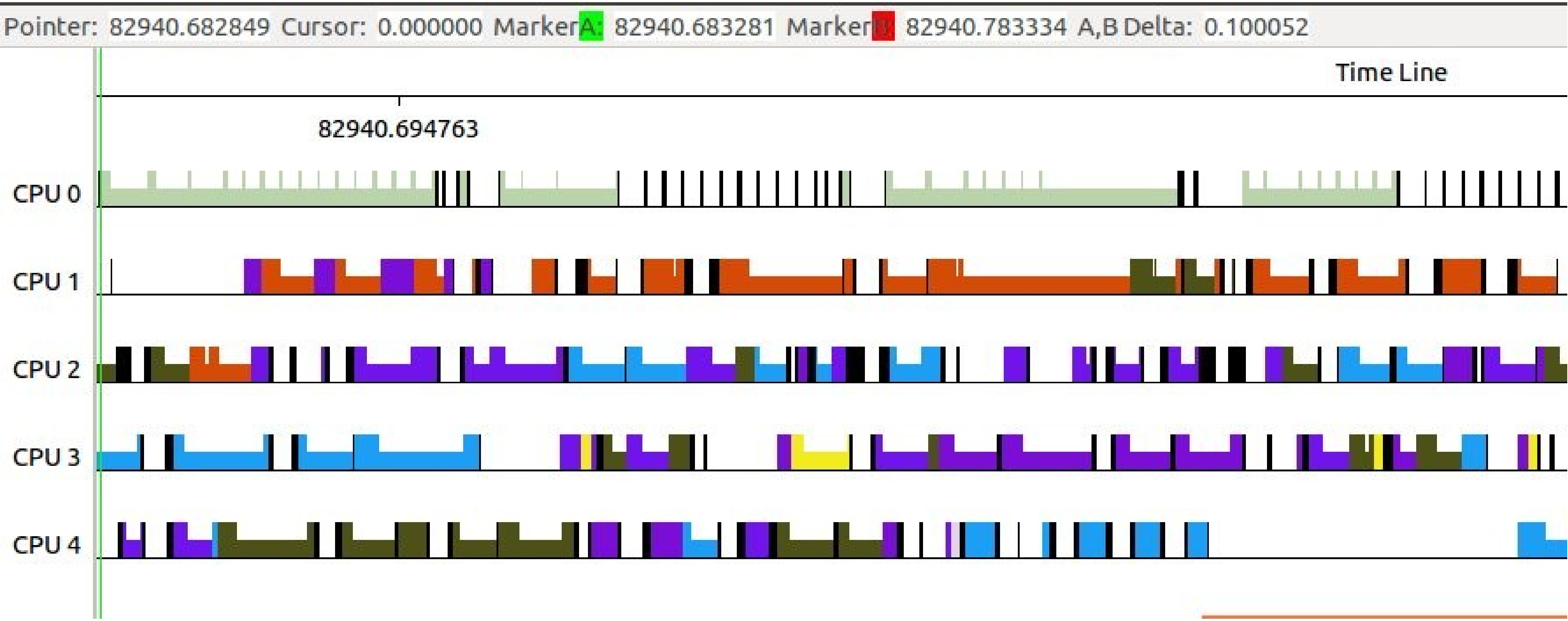
- trace-cmd encapsulates ftrace
  - Package available for Ubuntu image
  - Installed in next android image

- Graphic viewer : kernelshark
  - Nice human readable interface
  - Package available for Ubuntu image

Start to trace

- Have a look at trace-cmd
  - Let make 1st traces

trace-cmd record



Pointer: 82940.682849 Cursor: 0.000000 MarkerA: 82940.683281 MarkerB: 82940.783334 A,B Delta: 0.100052

trace-cmd record -e all -o example0.dat sleep 2

- Huge activity...
- Tracing activity...

•What to trace ?

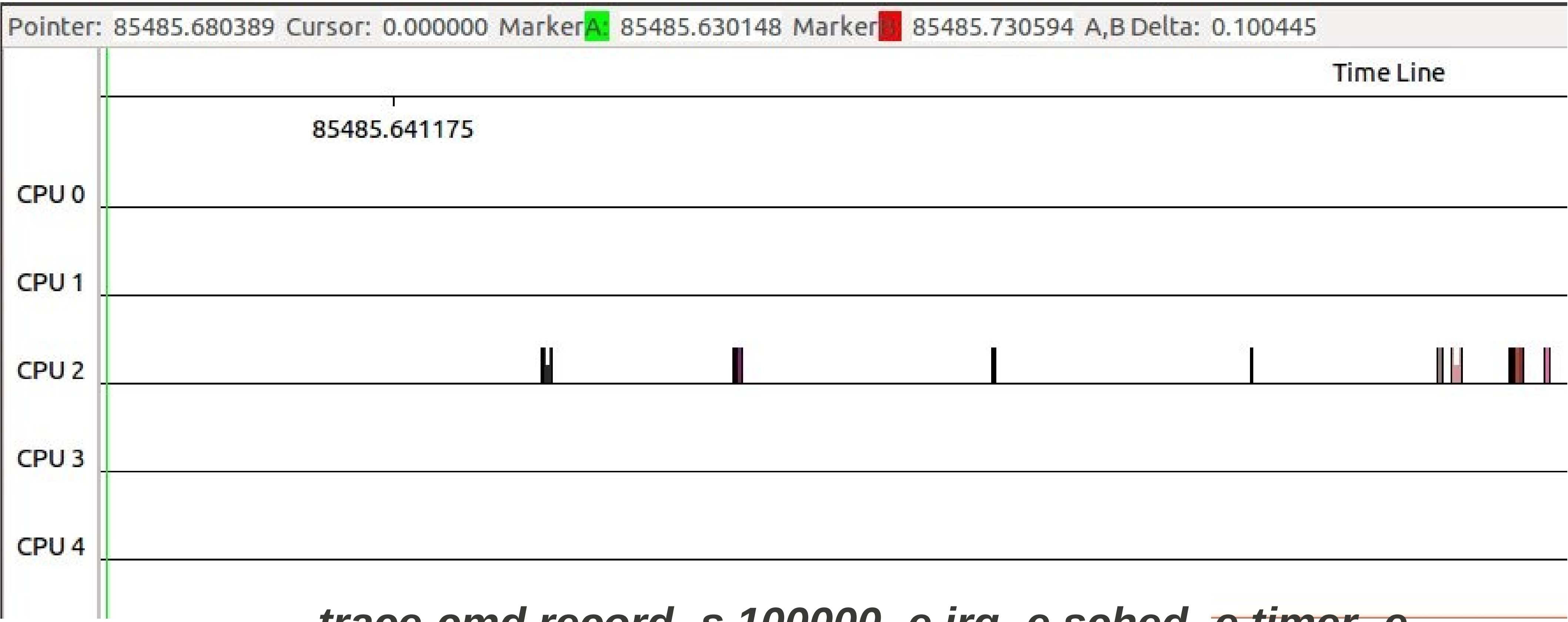
- Core state:
  - Idle state
  - Frequency scaling
- Wake up source:
  - Interruption
- Activity:
  - Sched
  - Timer
  - Workqueue

Ring buffer

- ftrace uses a ring buffer
  - Limited number of events

- trace-cmd record
  - Periodically read the ring buffer
  - -s option set the period (default is 1ms)
  - Generate “spurious” activity : 1 process per core
  - Trig deferrable activity

Sample period


Pointer: 85485.680389 Cursor: 0.000000 MarkerA: 85485.630148 MarkerB: 85485.730594 A,B Delta: 0.100445

trace-cmd record -s 100000 -e irq -e sched -e timer -e workqueue -o example1.dat sleep 2

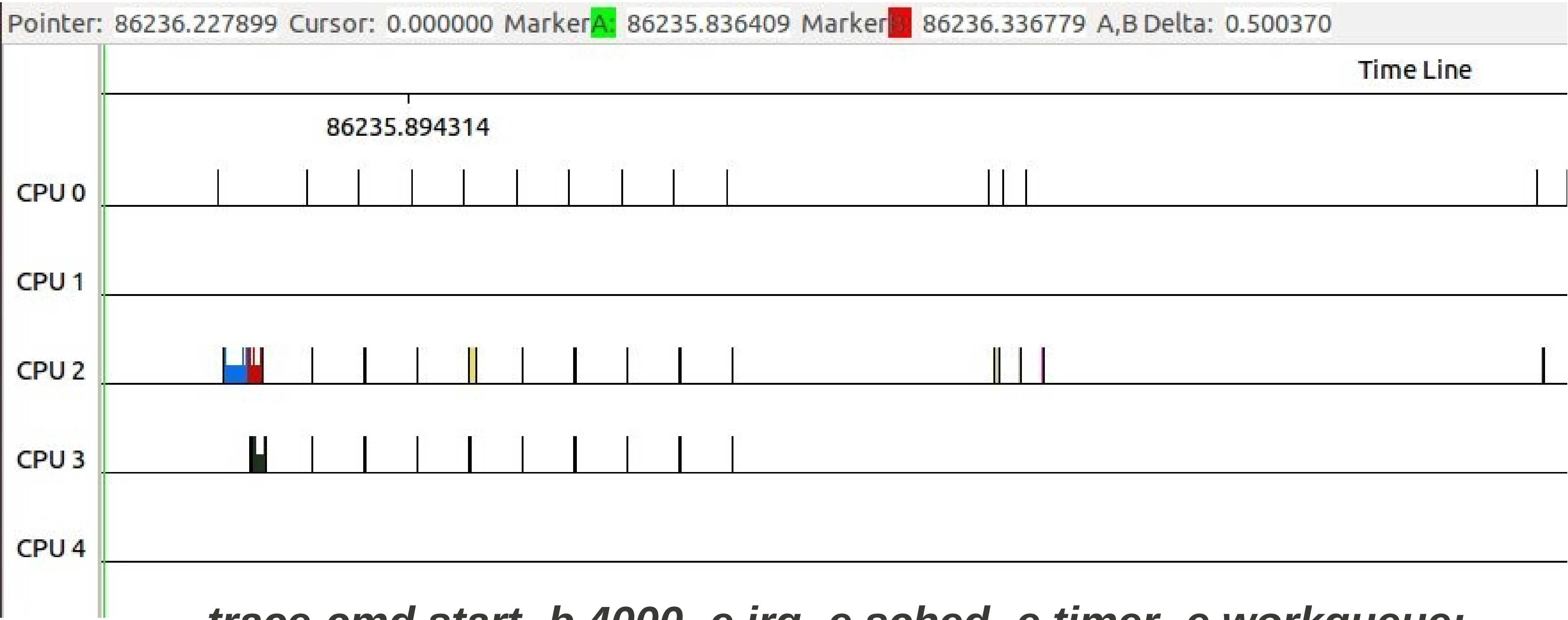
- Less activities ...
- Tracing activity ...

Stay quiet

- trace-cmd start / stop / extract
  - No noise activity during the record
  - circular buffer
  - Increase the buffer's size

- Make a new trace

Quiet trace



Pointer: 86236.227899 Cursor: 0.000000 MarkerA: 86235.836409 MarkerB: 86236.336779 A,B Delta: 0.500370

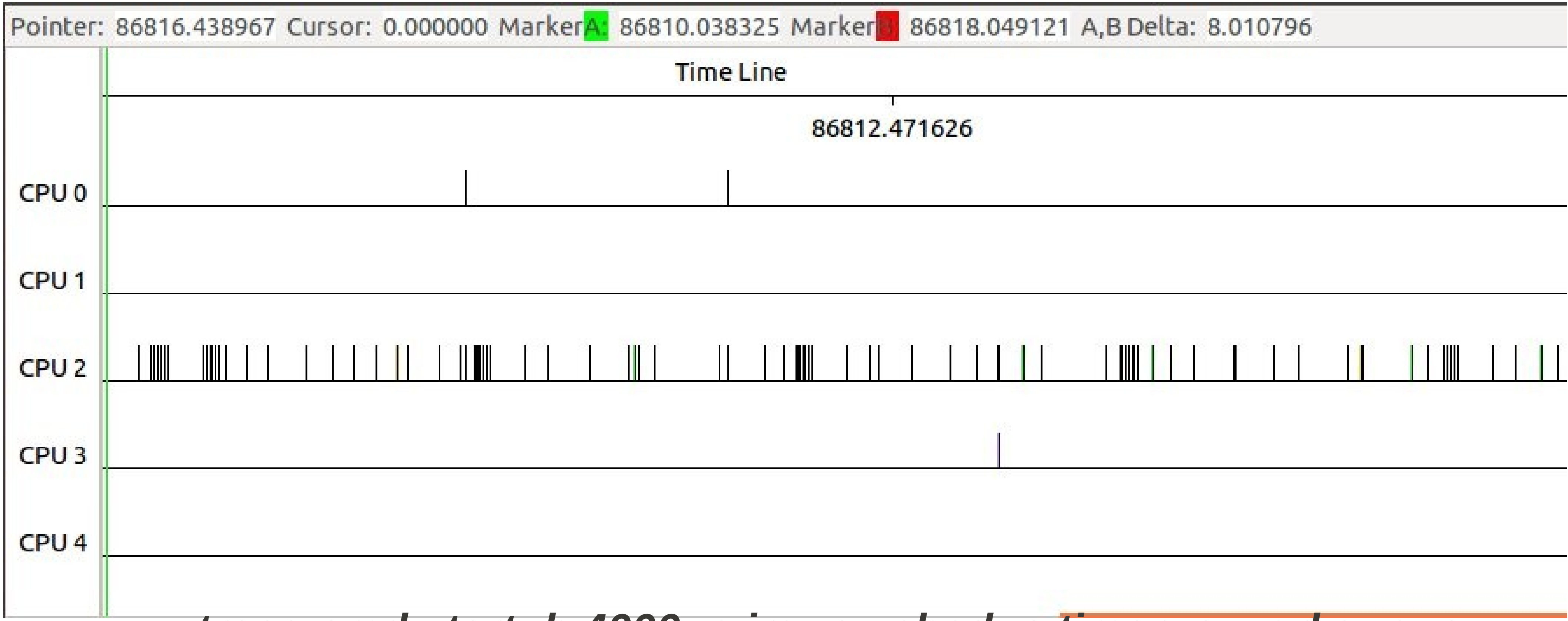
trace-cmd start -b 4000 -e irq -e sched -e timer -e workqueue; sleep 2; trace-cmd stop; trace-cmd extract -o example2.dat

- few activity...
- IRQ on big → CPU0

IRQ affinity

- The real wake up source
- Influence where task will run
- Use 1st CPU of the mask
  - CPU0 → big core
- Change the affinity
  - /proc/irq/*/smp_affinity
- irqbalance daemon
- Set default affinity on LITTLE

IRQ affinity



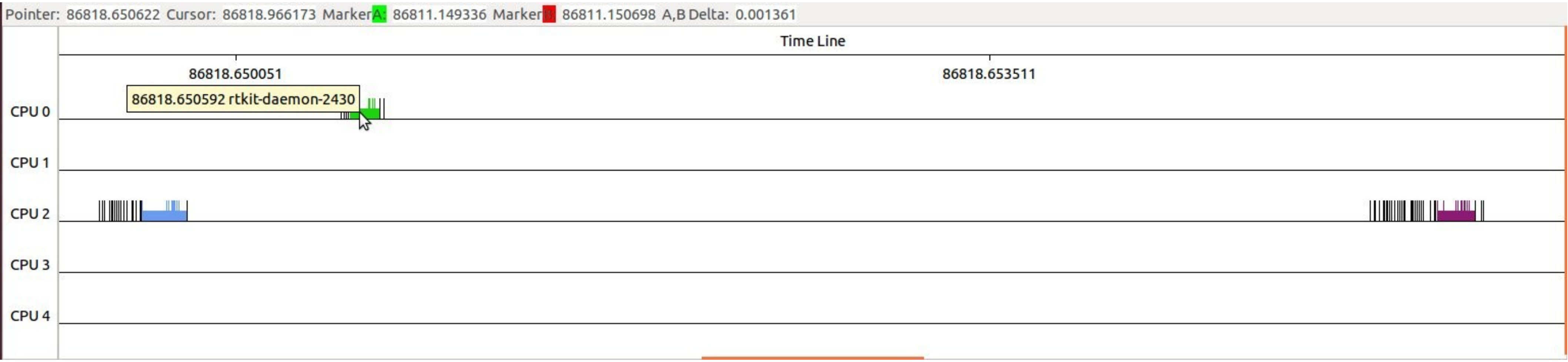
Pointer: 86816.438967 Cursor: 0.000000 MarkerA: 86810.038325 MarkerB: 86818.049121 A,B Delta: 8.010796

```
trace-cmd start -b 4000 -e irq -e sched -e timer -e workqueue;
sleep 2; trace-cmd stop; trace-cmd extract  -o example3.dat
```

- Nearly nothing on big cores...

RT tasks

- On previous trace
  - Only some RT tasks on big
- There is not much we can do
  - Set task affinity with taskset



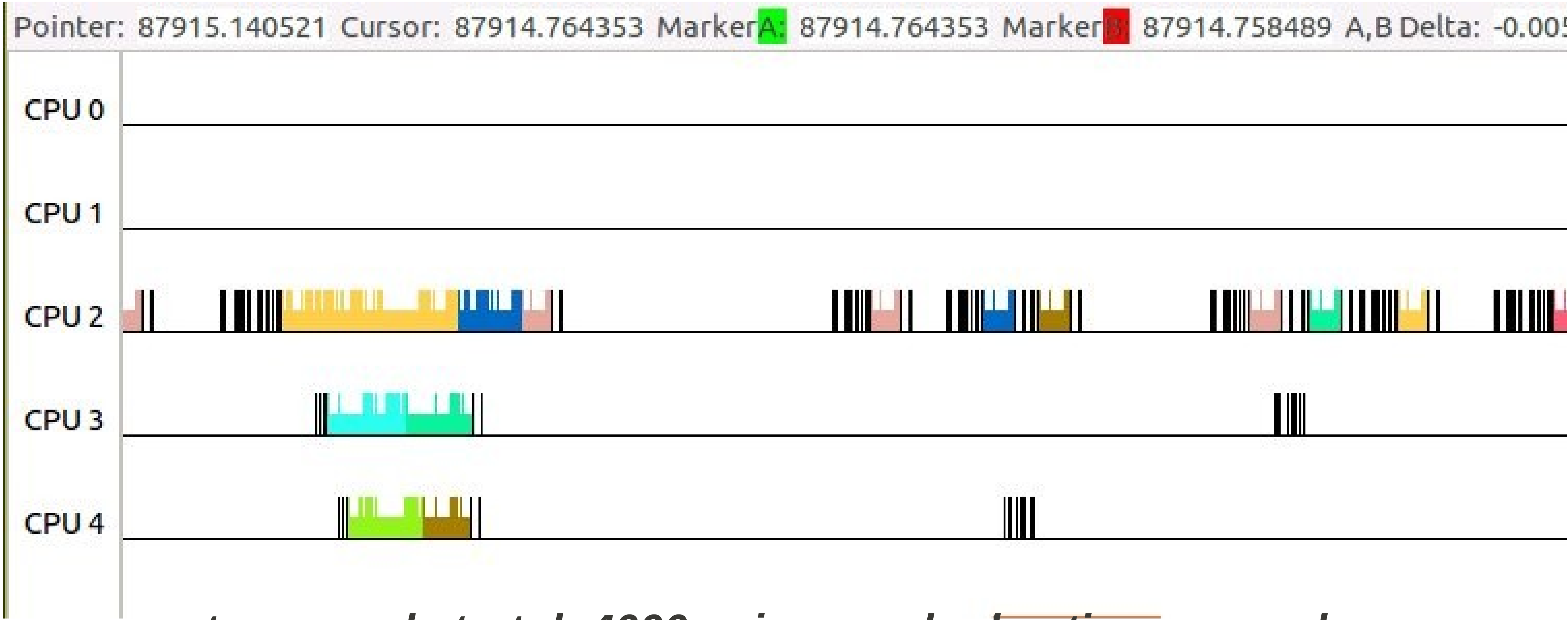
Pointer: 86818.650622 Cursor: 86818.966173 MarkerA: 86811.149336 MarkerB: 86811.150698 A,B Delta: 0.001361

86818.650592 rtkit-daemon-2430

Low CPU load

- Idle is almost aligned with what we want
- Test behavior with low load tasks
- Cyclictest
  - Create small tasks that wake up periodically

cyclictest



Pointer: 87915.140521 Cursor: 87914.764353 MarkerA: 87914.764353 MarkerB: 87914.758489 A,B Delta: -0.005

```
trace-cmd start -b 4000 -e irq -e sched -e timer -e workqueue
-e cpu_idle; cyclictest -t 7 -q -D 1; trace-cmd stop; trace-cmd
extract  -o exampl4.dat
```

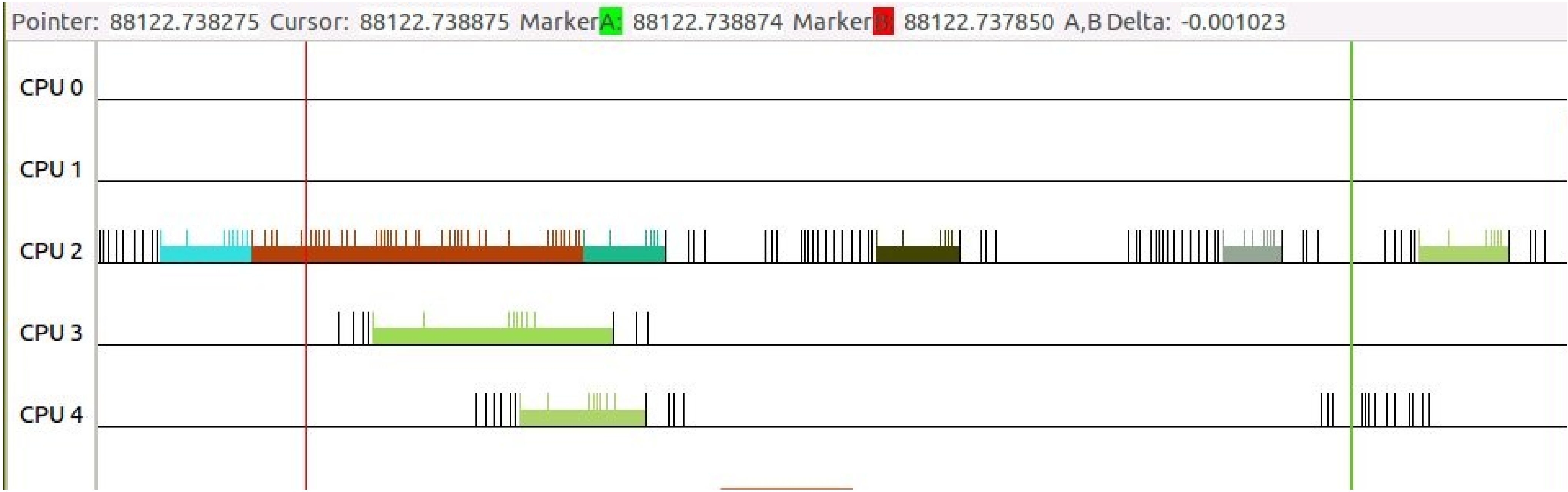
- Stay on LITTLE cores

Small task packing

- Most of cyclictest thread on CPU2
  - Some on CPU3
  - Few on CPU4

- Why ?
  - Pack buddy CPU is busy
  - Go back to default behavior
  - Worth to parallelize in a cluster

Deeply in the trace



Pointer: 88122.738275 Cursor: 88122.738875 MarkerA: 88122.738874 MarkerB: 88122.737850 A,B Delta: -0.001023

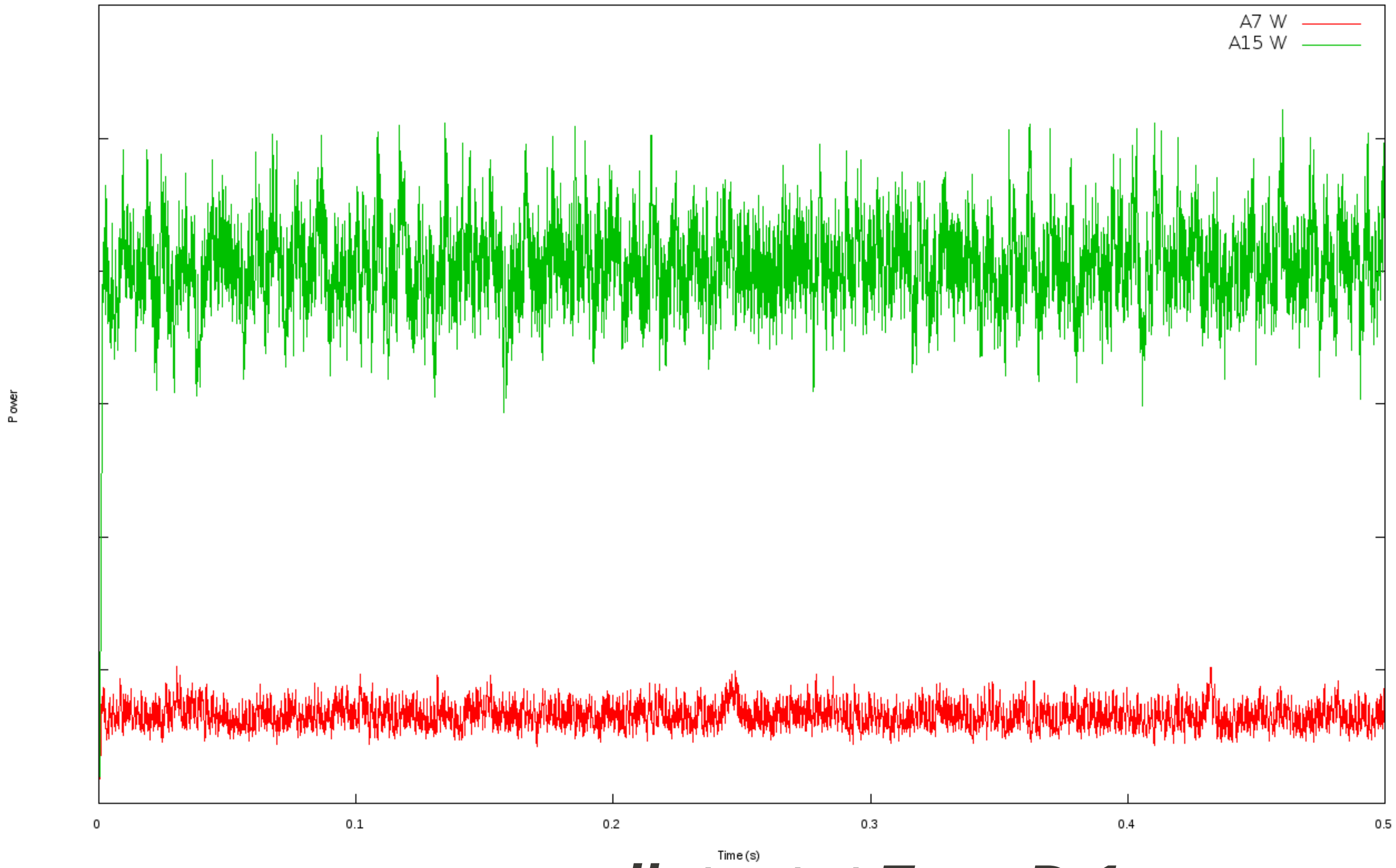
- Packing task → green line
- Spreading task → red line

Power consumption

- Measure power of the use case
  - ARM-probe HW
  - Command line SW for acquisition
    - Thanks to Andy Green
  - Gnuplot for displaying

- Make a measure

1st power result



cyclictest -t 7 -q -D 1

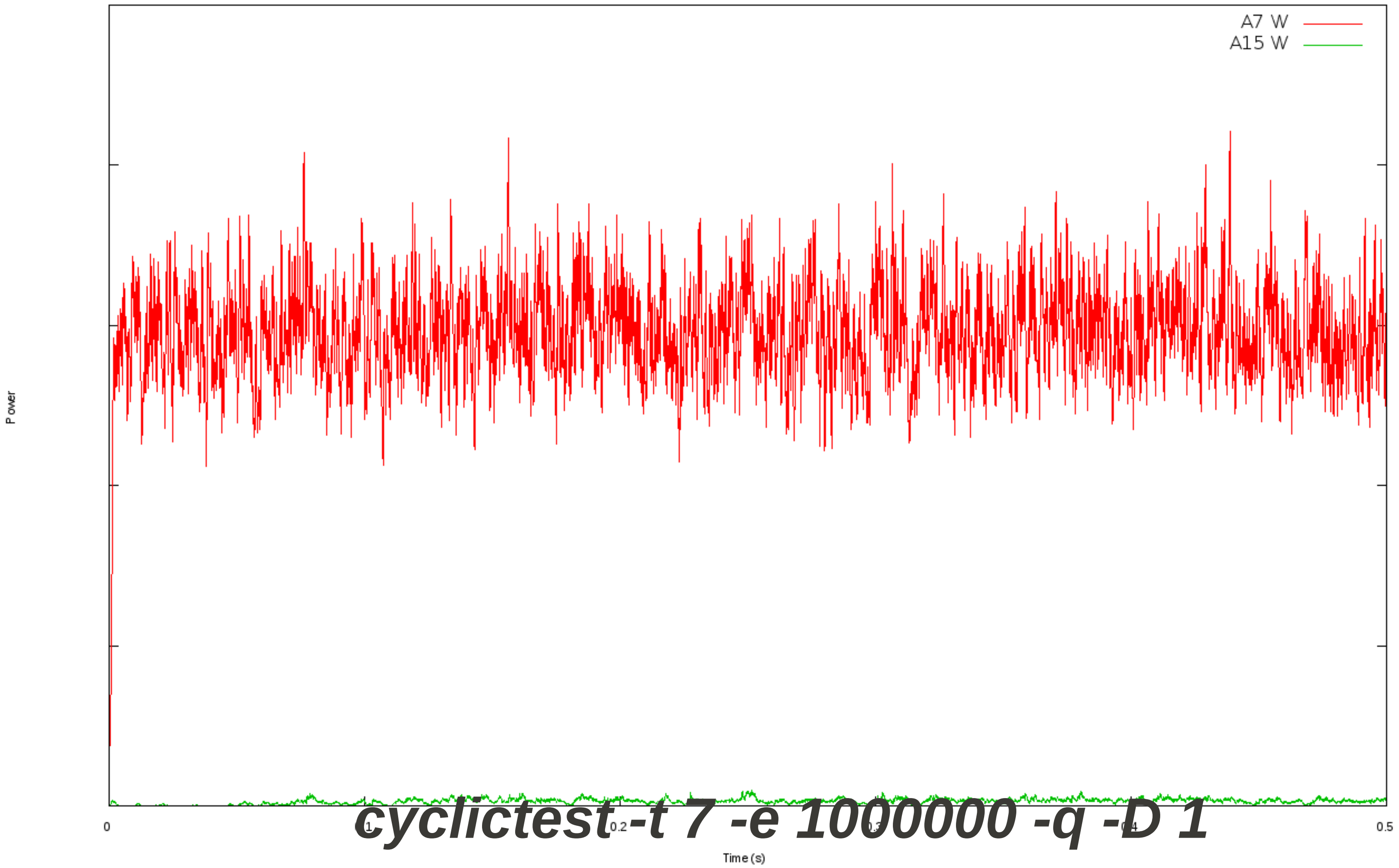
- Both cluster are always on and consuming !!!

cyclictest

- Set the cpu_dma_latency QoS
  - Used by cpuidle
  - -e option set the QoS value

- Make another measure

New power result



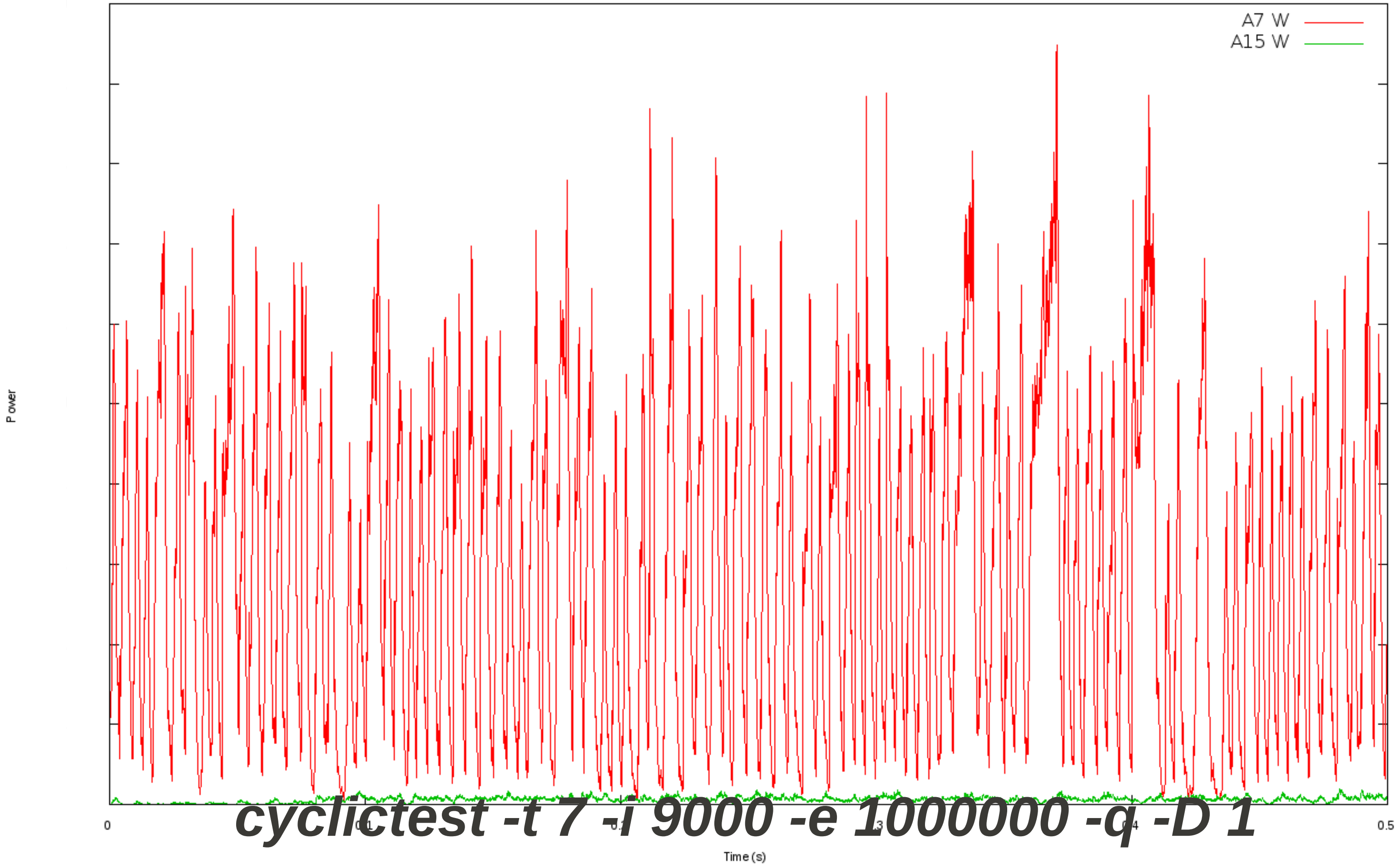
cyclictest -t 7 -e 1000000 -q -D 1

- Big cluster is off
- LITTLE cluster is on

cyclictest

- Cluster goes down iff idle duration >= 1ms
- Default cyclictest interval is 1ms
  - Can't enter off state
  - -i option increases the interval
- Make another measure

Power measure



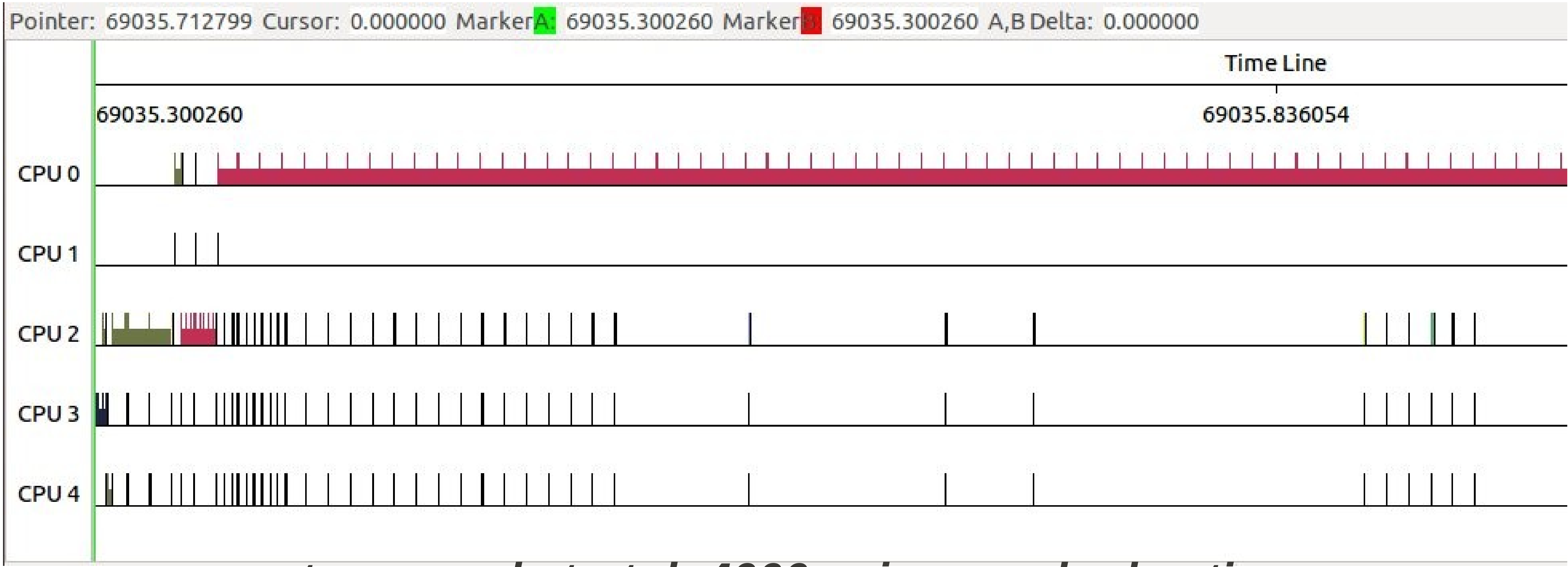
cyclictest -t 7 -i 9000 -e 1000000 -q -D 1

- Big cluster is off
- LITTLE cluster toggles

Heavy task

- Small tasks are packed
- What about heavy task ?
- Sysbench
  - --test=cpu : prime number computation
  - --test=threads : lock thread

sysbench

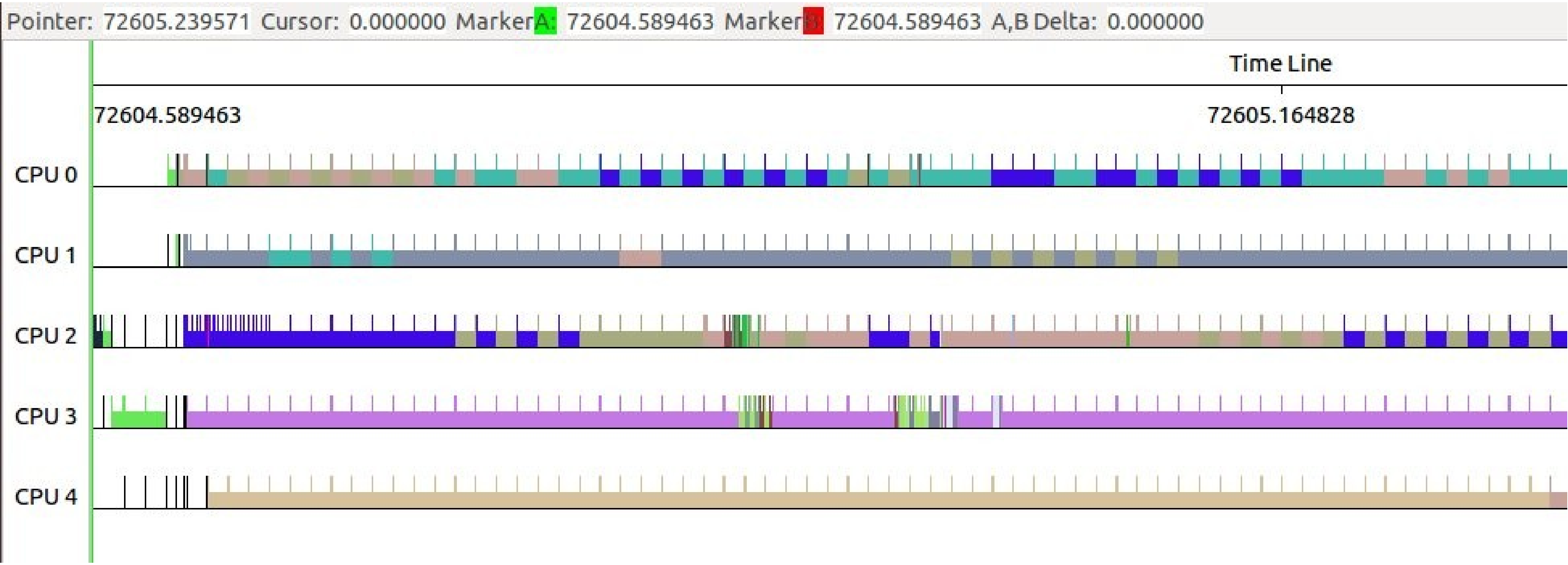


Pointer: 69035.712799 Cursor: 0.000000 MarkerA: 69035.300260 MarkerB: 69035.300260 A,B Delta: 0.000000
Time Line
69035.300260
69035.836054
CPU 0
CPU 1
CPU 2
CPU 3
CPU 4

trace-cmd start -b 4000 -e irq -e sched -e timer -e workqueue -e cpu_idle; sysbench –test=cpu –num-thread=1 –max-time=1 run; trace-cmd stop; trace-cmd extract -o exampl4.dat

- Move on big core

sysbench



Pointer: 72605.239571 Cursor: 0.000000 MarkerA: 72604.589463 MarkerB: 72604.589463 A,B Delta: 0.000000

```
trace-cmd start -b 4000 -e irq -e sched -e timer -e workqueue -e
cpu_idle; sysbench –test=cpu –num-thread=7 –max-time=1 run;
trace-cmd stop; trace-cmd extract  -o exampl8.dat
```

- Move on big core

taskset

- Pin a task on a subset of CPU
  - Check what is the best scheduling behavior
- Performance
  - Fastest configuration
- Powersaving
  - Most thrifty configuration

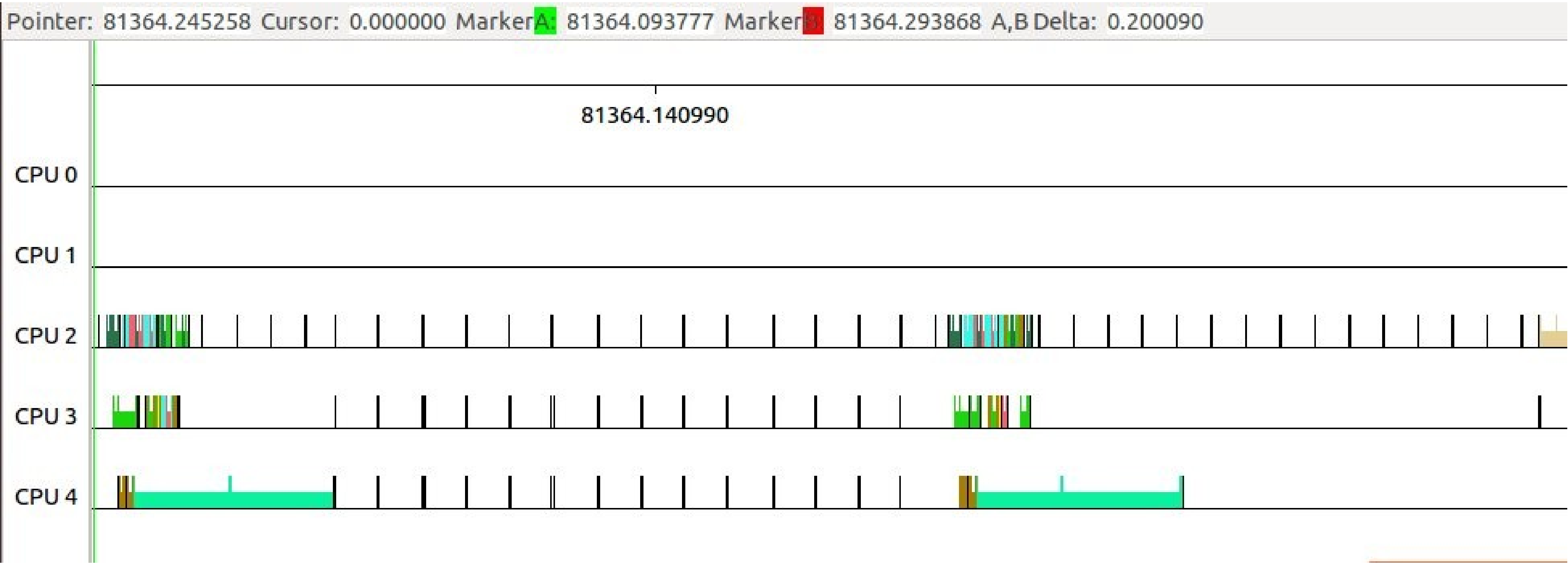
taskset

- Previous test
  - Better to put everything on big ?

Real use cases

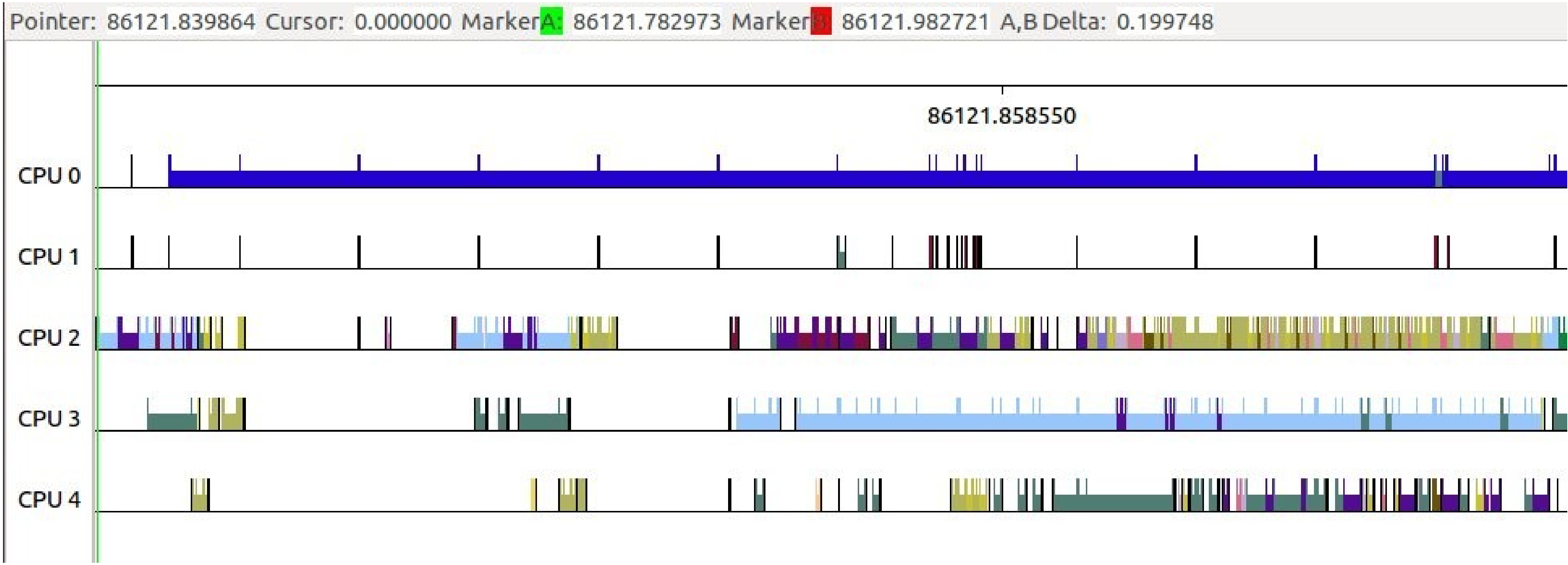
- MP3 playback
- Web browsing

MP3 playback



Pointer: 81364.245258 Cursor: 0.000000 MarkerA: 81364.093777 MarkerB: 81364.293868 A,B Delta: 0.200090

Web browsing



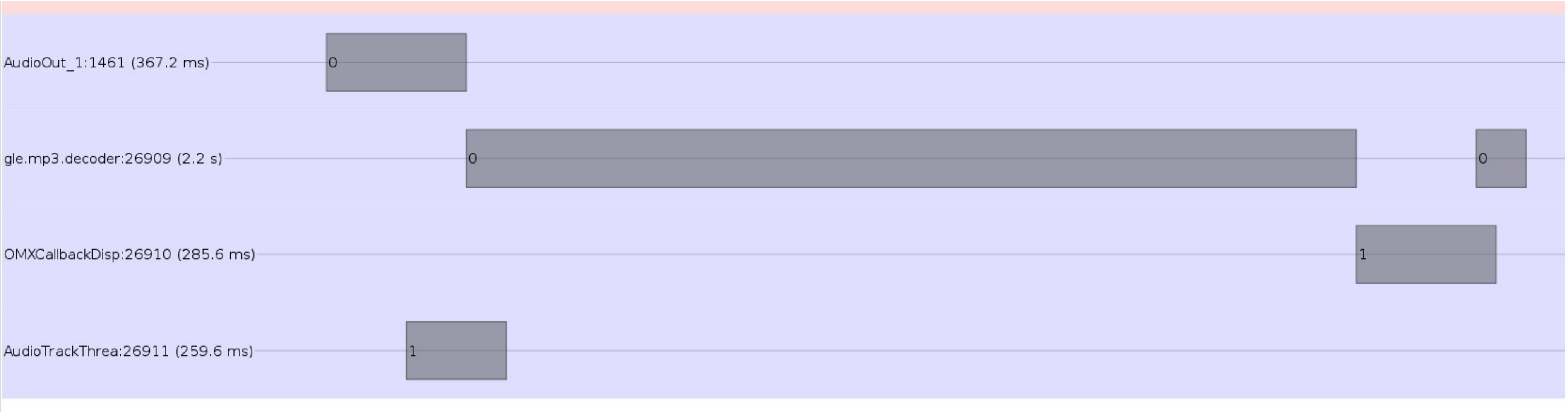
Pointer: 86121.839864 Cursor: 0.000000 MarkerA: 86121.782973 MarkerB: 86121.982721 A,B Delta: 0.199748

Question ?

Thank you

Backup slide

- MP3 sequence


AudioOut_1:1461 (367.2 ms)
gle.mp3.decoder:26909 (2.2 s)
OMXCallbackDisp:26910 (285.6 ms)
AudioTrackThrea:26911 (259.6 ms)