Experiments report

Francis Giraldeau

2016-05-05

École Polytechnique de Montréal

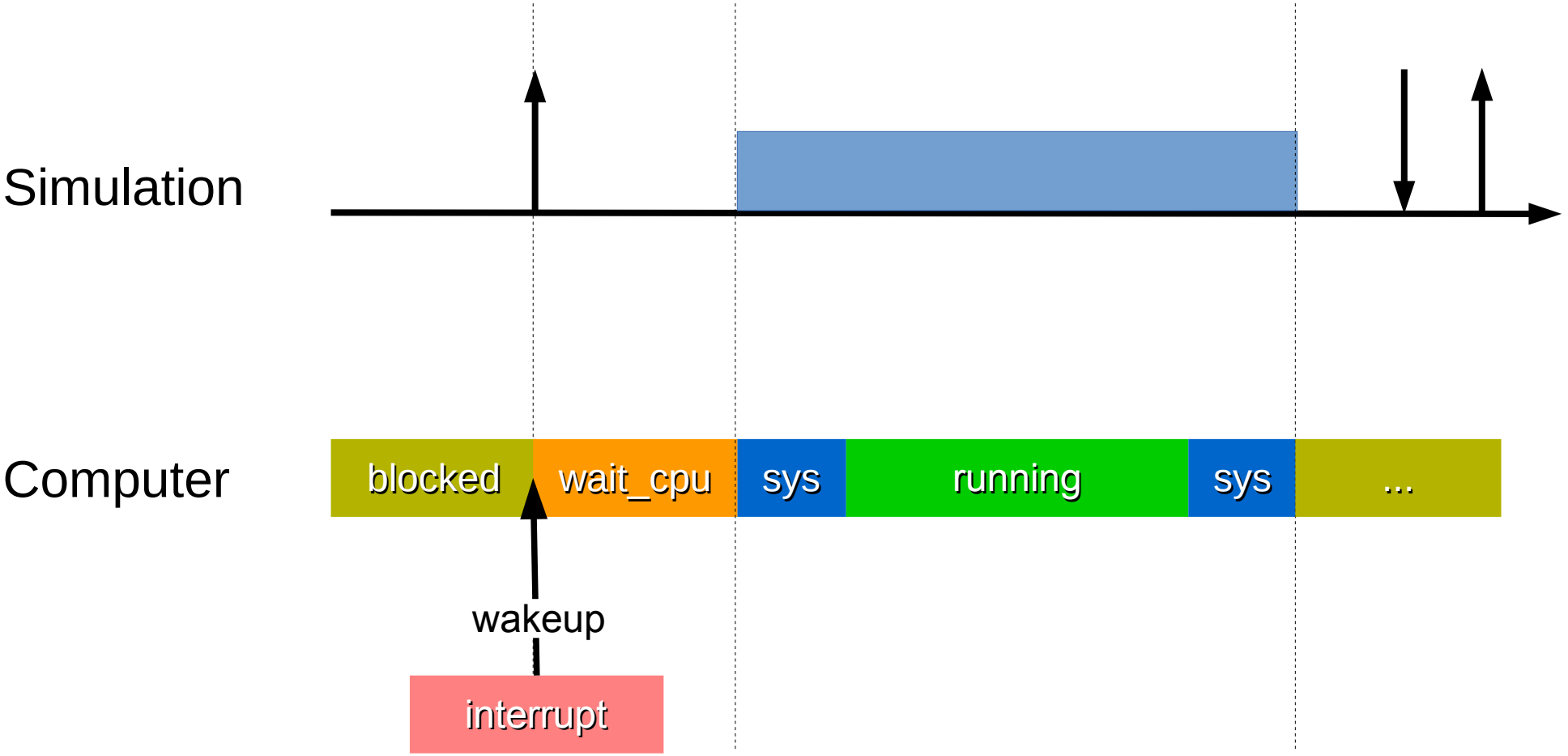
Laboratoire **DORSAL**

Outline

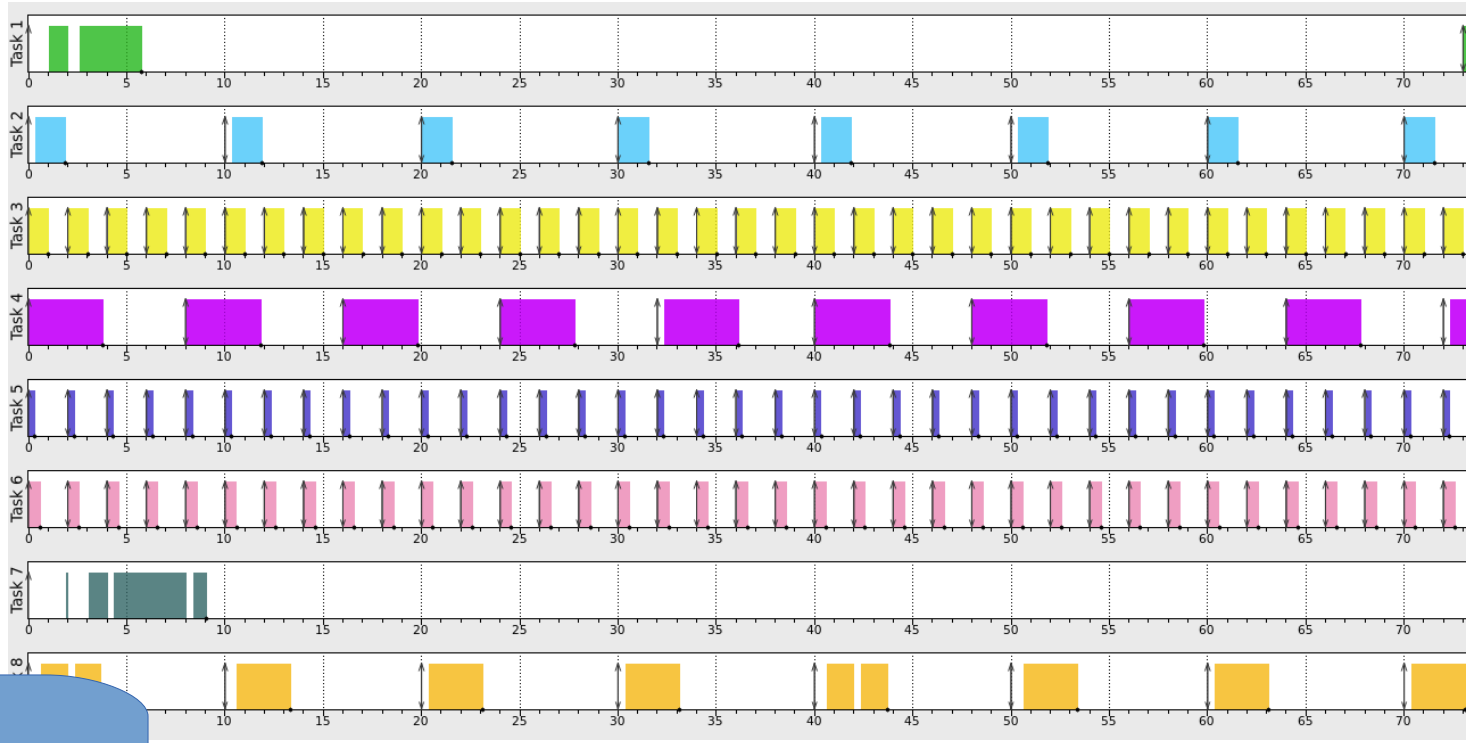
- Real-time scheduling simulation compared to actual execution
- Real-time latency debugging
- Arduino monitor for GPIO testing
- Function graph tracing with LTTng

Scheduling simulation (1)

- Study Linux SCHED_DEADLINE behavior
  - Parameters: runtime, deadline, period
  - Linux standard (with or without PREEMPT_RT)
- SimSo simulator
- rt-app testing tool + kernel tracing
- Task set randomly generated

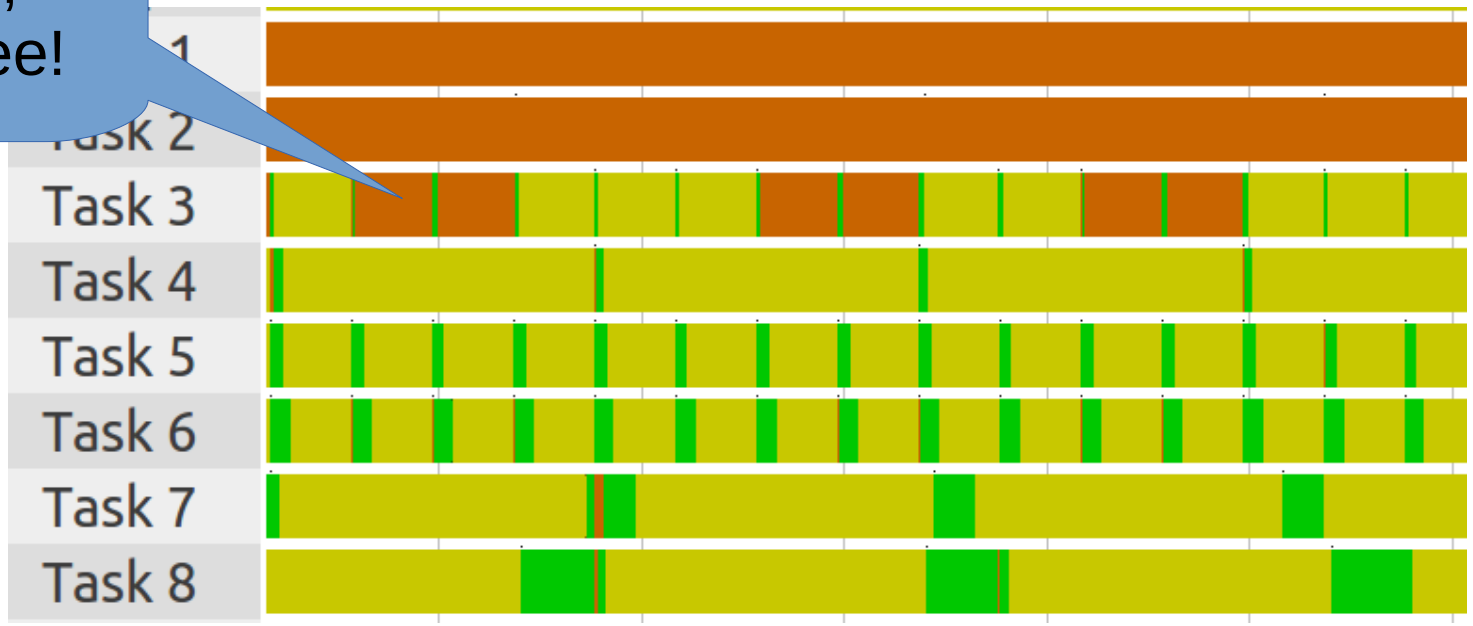

Simulation
Computer
blocked
wait_cpu
sys
running
sys
...
wakeup
interrupt

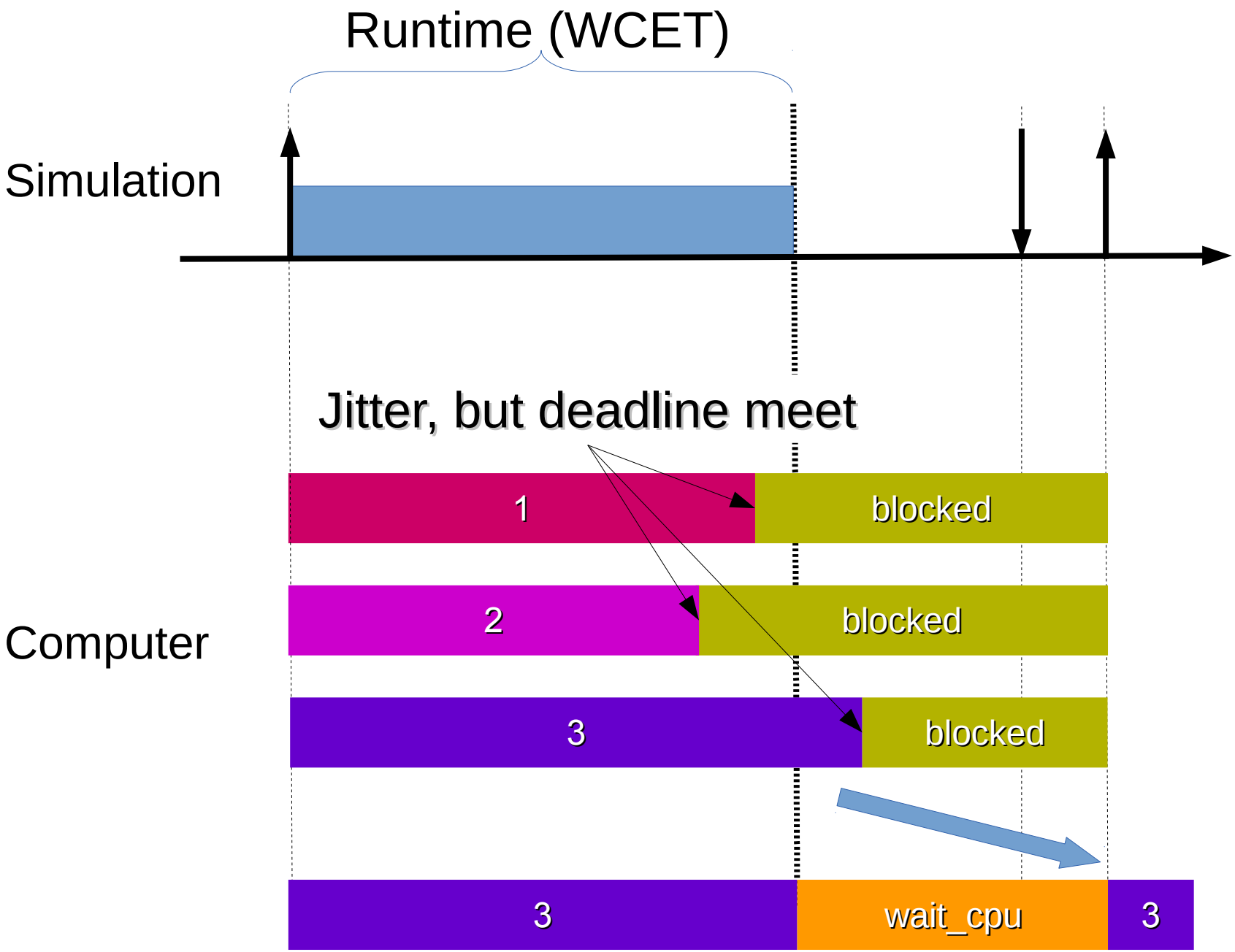
Simulation

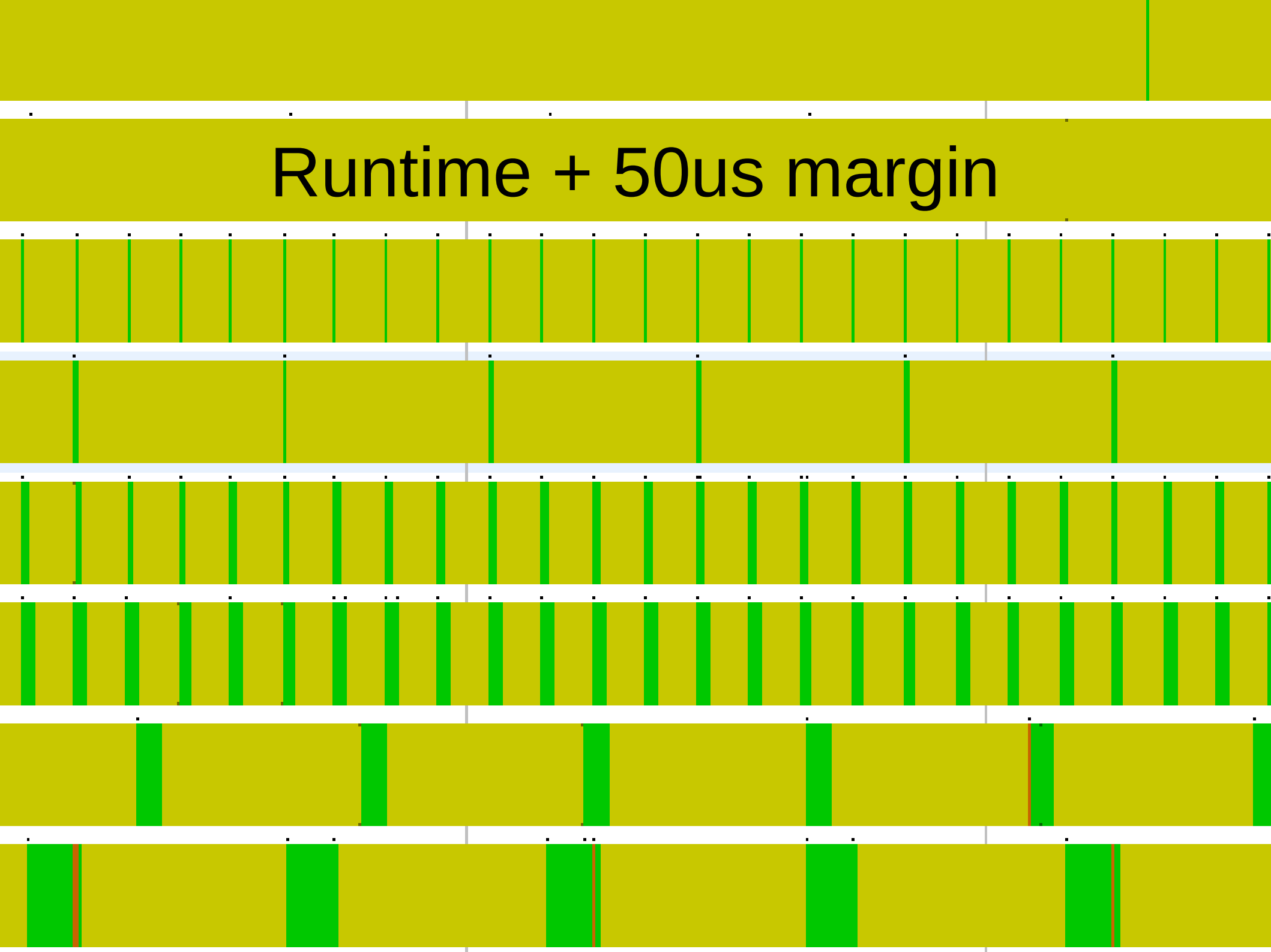


Wait for cpu,
but CPU is free!

Computer




Runtime (WCET)
Simulation
Jitter, but deadline meet
Computer
1
blocked
2
blocked
3
blocked
3
wait_cpu
3


Runtime + 50us margin

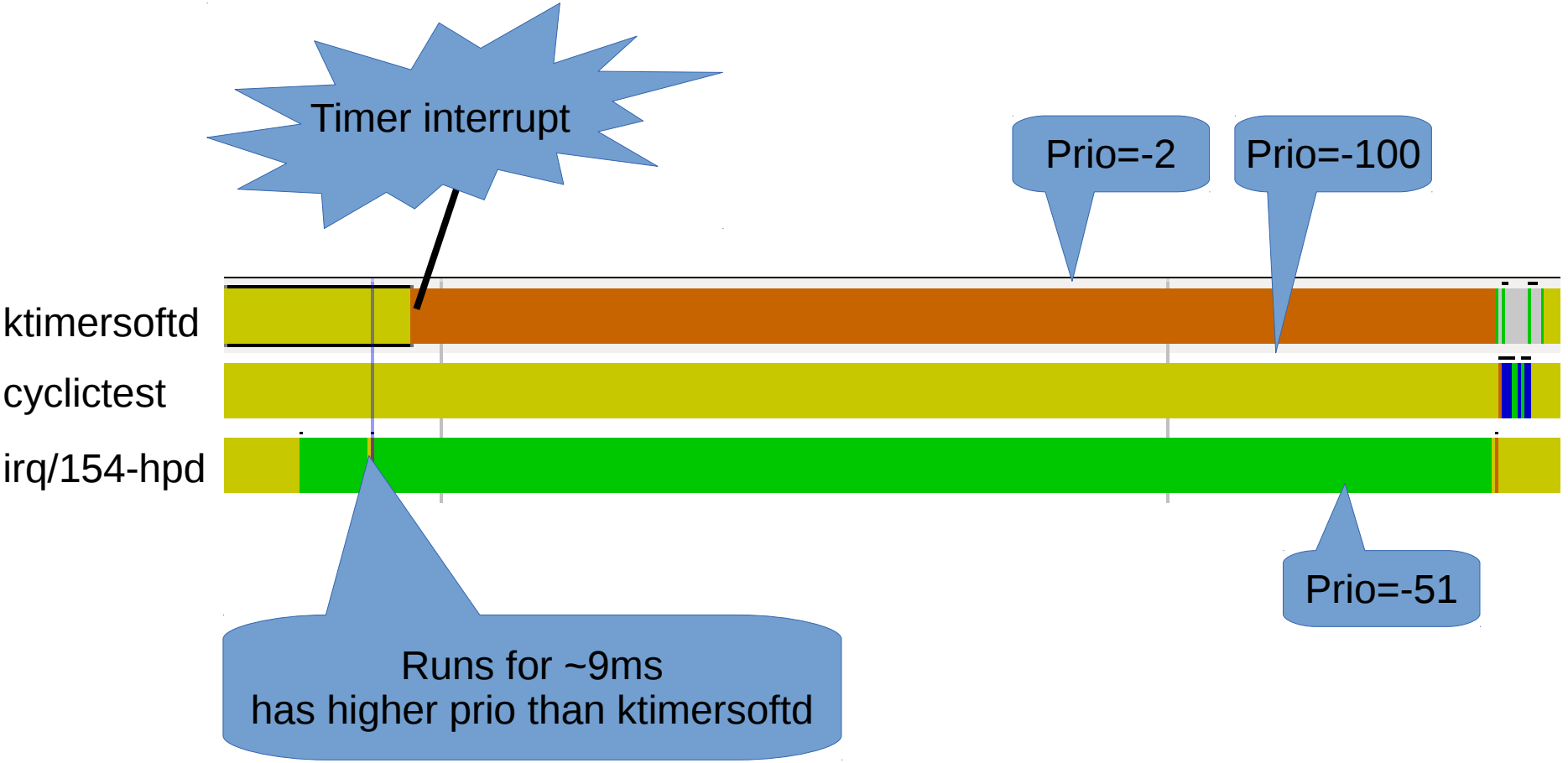
Real-time latency debugging (1)

- Jetson TK1
- Linux 4.4 PREEMPT_RT
- Cyclictest instrumentation with LTTng
  - Kernel and user space trace
  - Trigger snapshot on high wake-up latency

Real-time latency debugging (2)

```
/* Configure tracing */
lttng_create_session_snapshot(SESSNAME, "file:///tmp/cyclictest");
handle = lttng_create_handle(SESSNAME, &dom);
lttng_enable_channel(handle, &chan);

lttng_enable_event(handle, &ev, CHANNAME);
lttng_add_context(handle, &ctx, "*", CHANNAME);

lttng_start_tracing(SESSNAME);


/* Snapshot */
pthread_cond_wait(&cond, &mutex);

lttng_stop_tracing(SESSNAME);
out = lttng_snapshot_output_create();
lttng_snapshot_record(SESSNAME, out, 0);
lttng_snapshot_output_destroy(out);
lttng_start_tracing(SESSNAME);
```

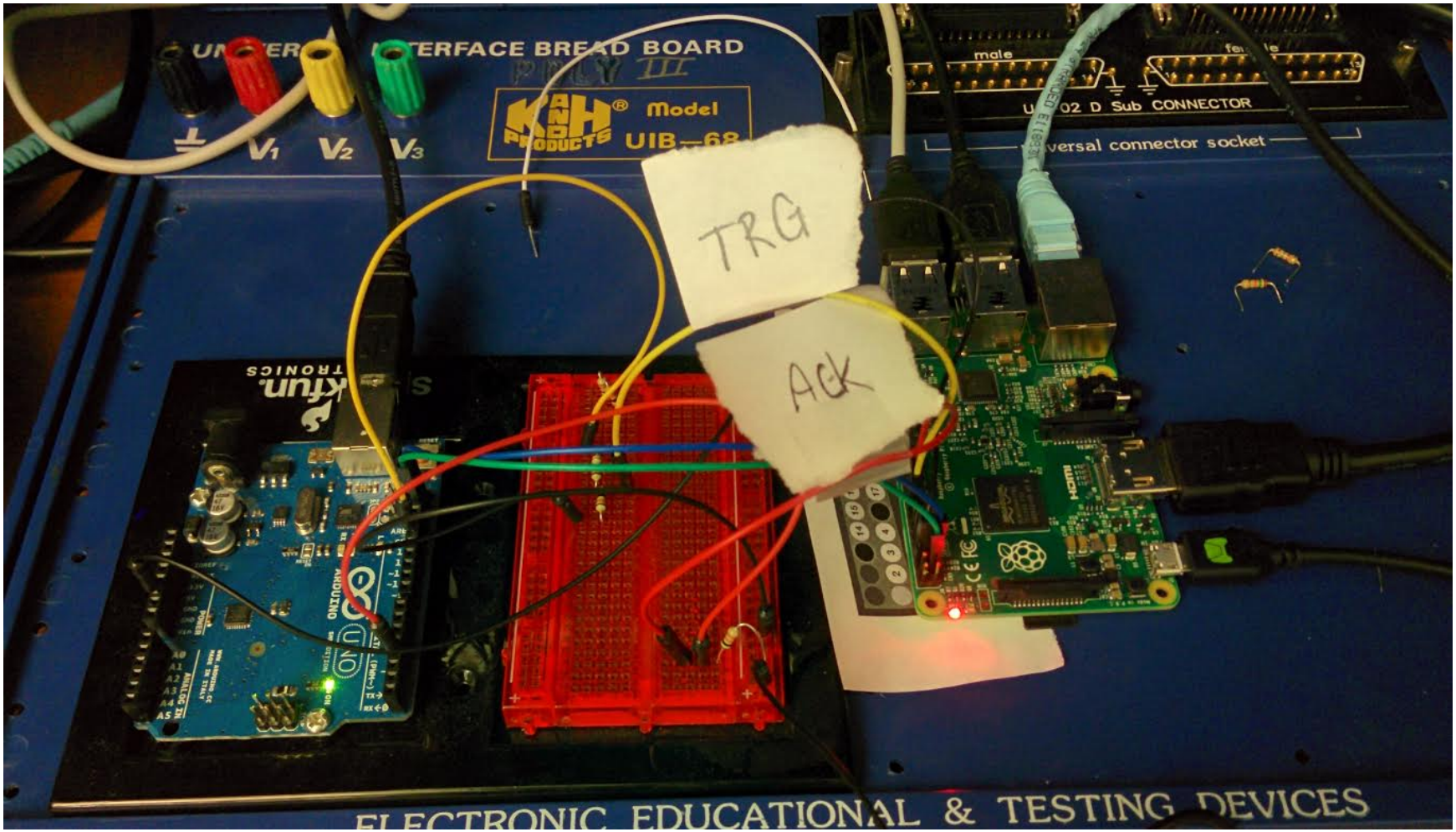
Real-time latency debugging (3)



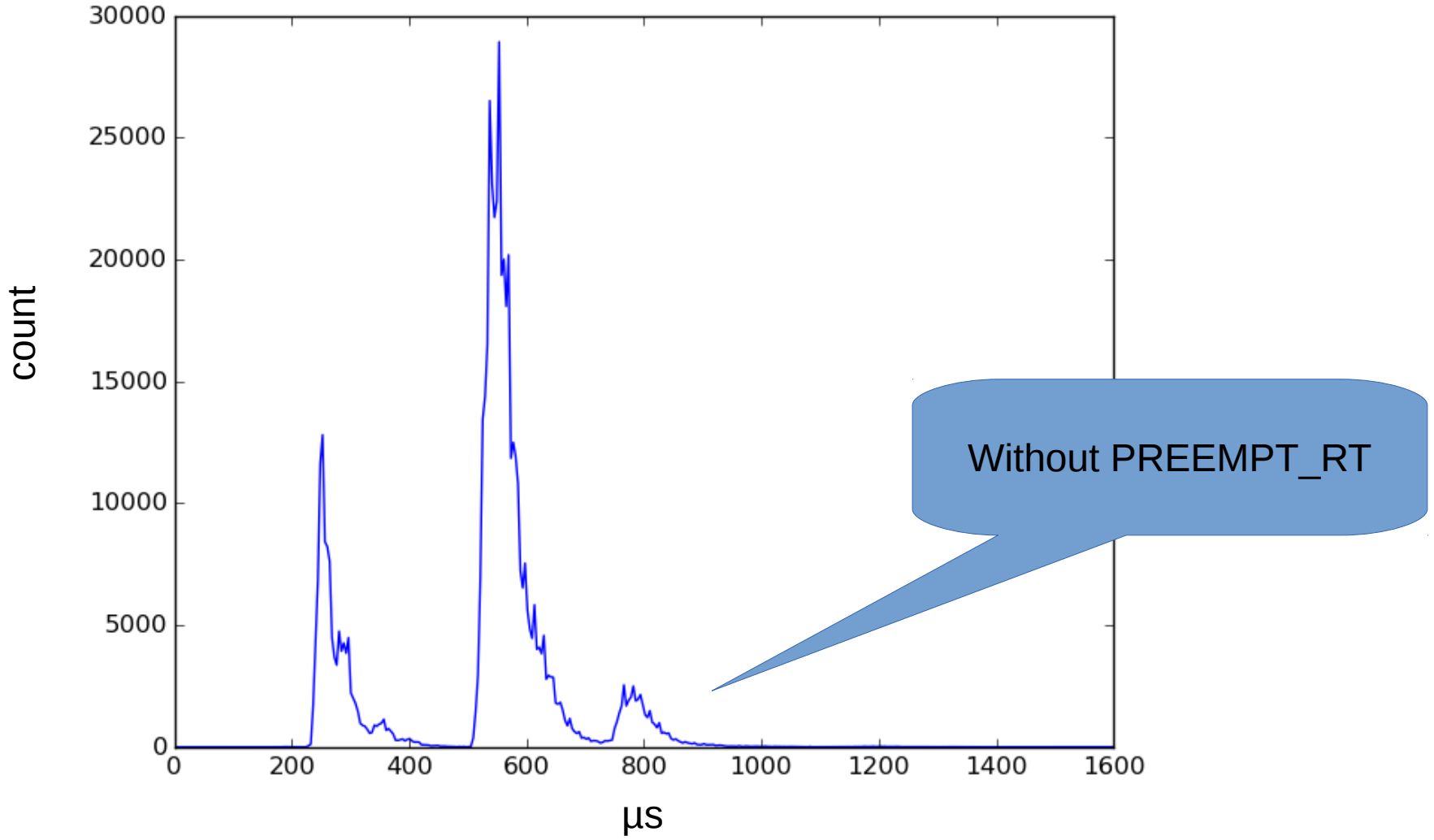
Arduino monitor (1)

- Black-box measure of interrupt acknowledgement
- Arduino
  - Raises output pin and measure the time for the acknowledge pin to go high
  - Build latency histogram
- RaspberryPI
  - Wait for GPIO trigger and acknowledge
  - Control Arduino and download histogram over i2c

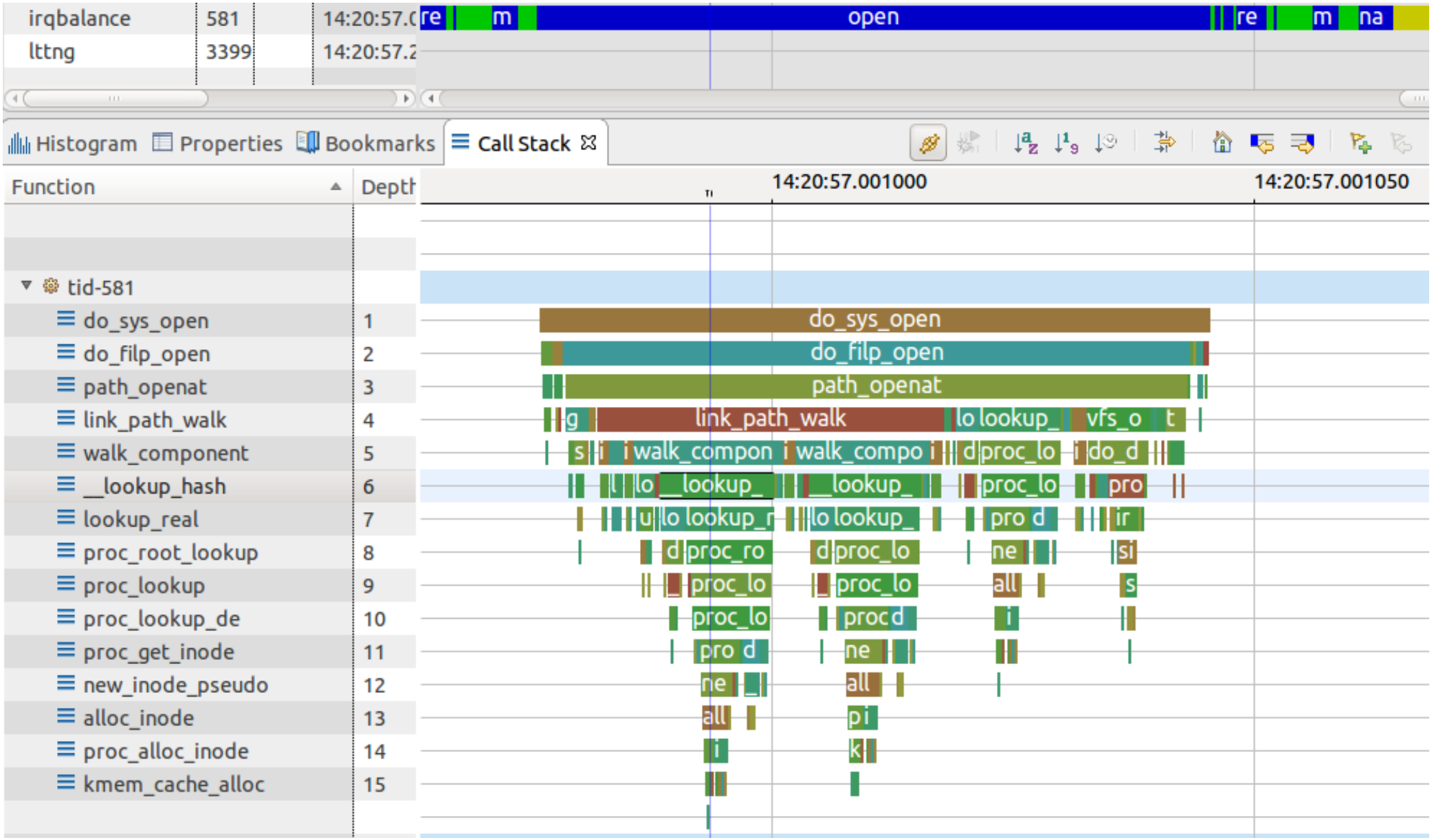
Arduino monitor (2)



Arduino monitor (3)
GPIO round-trip latency histogram



Ftrace function graph with LTTng



Thanks to Professor Michel Dagenais and our partners EfficiOS and Ericsson.

Home page: http://step.polymtl.ca/~fgiraldeau/

Software: https://github.com/giraldeau