1. Selamawit Tegegn(sntegegn@wustl.edu) , Leiquan Pan(lpan22@wustl.edu), yiying Lu(lu.yiying@wustl.edu)

**"Module Design"**

*We used the module\_param macro to define two parameters log\_sec and log\_nsec. Each of these parameters have an unsigned long data type and 0000 file permission. In the case where the module failed to provide the parameters, we initialized the parameters to a default value of 1sec*.

**"Timer Design and Evaluation"**

Implement requirement:

1. *In our module init function,*

* *we initialized and started our hrtimer.*

*hrtimer\_init( &hrtimers, CLOCK\_MONOTONIC, HRTIMER\_MODE\_REL )*

*hrtimer\_start( &hrtimers, kt, HRTIMER\_MODE\_REL )*

* *we defined a call back function that runs after the timer expires.*

*hrtimers.function = &timer\_callback;*

* *we used the module parameters to define the ktime variable that represents the first timeout value of the timer.*

*ktime\_t kt;*

*kt = ktime\_set( log\_sec, log\_nsec)*

*(2) In the function that runs after the timer expires we called an hrtimer\_forward function to reschedule the timer’s next expiration.*

*hrtimer\_forward( timer\_for\_restart, currtime , ktimes )*

*Since we want the timer to be restarted we return HRTIMER\_RESTART.*

*(3) In our module exit, we used hrtimer\_cancel to terminate the timer*

*hrtimer\_cancel( &hrtimers )*

Result:

(when interval is 0.1)

[ 922.637998] log\_sec:0,log\_nsec:100000000  
[ 922.737987] krtimer restart.  
[ 922.837990] krtimer restart.  
[ 922.937986] krtimer restart.  
[ 923.037986] krtimer restart.  
  
(when interval is 1.1)

[14298.416857] log\_sec:1,log\_nsec:100000000

[14299.516853] krtimer restart.

[14300.616856] krtimer restart.

[14301.716856] krtimer restart.

[14302.816861] krtimer restart.

(when interval is 2.1)

[14464.075449] log\_sec:2,log\_nsec:100000000

[14466.175455] krtimer restart.

[14468.275456] krtimer restart.

[14470.375481] krtimer restart.

[14472.475464] krtimer restart.

[14474.575465] krtimer restart.

[14476.675470] krtimer restart.

[14478.775480] krtimer restart.

[14480.875477] krtimer restart.

(when interval is 3.1)

[14581.775338] log\_sec:3,log\_nsec:100000000

[14584.875337] krtimer restart.

[14587.975347] krtimer restart.

[14591.075349] krtimer restart.

[14594.175357] krtimer restart.

[14597.275359] krtimer restart.

[14600.375366] krtimer restart.

[14603.475369] krtimer restart.

[14606.575377] krtimer restart.

[14609.675379] krtimer restart.

*The module parameters are used to set the timeout value of the timer.*

We could tell that the interval equal to the the setting of our timer, which is equaled to log\_sce+log\_nsec.

**"Thread Design and Evaluation"**

1. *In our init function, we created and spawned a thread using kthread\_run*

*kthread\_run( thread\_func,"data for thread\_func","mythread")*

*(2) We also created a function that runs on one or more threads. This function prints out a message when it is invoked.*

*(3) We add a loop to our function, In the loop,*

*- we print out the number of voluntary and involuntary context switches that happen when the current kernel thread is running.*

*- the set\_current\_state function is used to change the state of the currently executing process from TASK\_RUNNING to TASK\_INTERRUPTIBLE. The task gets removed from the run queue and sleeps until woken up by another process or a signal.*

*set\_current\_state(TASK\_INTERRUPTIBLE)*

*- the schedule function allows the thread to voluntarily relinquish the processor.*

*schedule()*

*(4) We modified our timer callback function(from part 3) to wake up the sleeping kernel thread each time it is called.*

*wake\_up\_process(&task\_pointer)*

*(5) In the module’s exit function, kthread\_stop is used to notify the thread that it needs to stop executing.*

*kthread\_stop( task\_pointer )*

(when interval is 0.1)  
[14999.536610] simple module initialized

[14999.536623] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0

[14999.636633] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0

[14999.736620] Another iteration of the loop has started,the value of nvscw: 3 , the value of nivcsw: 0

[14999.836620] Another iteration of the loop has started,the value of nvscw: 4 , the value of nivcsw: 0

[14999.936619] Another iteration of the loop has started,the value of nvscw: 5 , the value of nivcsw: 0

(when interval is 1.1)

[15074.497865] simple module initialized

[15074.497871] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0

[15075.597889] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0

[15076.697885] Another iteration of the loop has started,the value of nvscw: 3 , the value of nivcsw: 0

[15077.797887] Another iteration of the loop has started,the value of nvscw: 4 , the value of nivcsw: 0

[15078.897893] Another iteration of the loop has started,the value of nvscw: 5 , the value of nivcsw: 0

[15079.743987] The thread is terminating

[15079.744075] simple module is being unloaded

(when interval is 2.1)

[15109.888316] simple module initialized

[15109.888497] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0

[15111.988332] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0

[15114.088329] Another iteration of the loop has started,the value of nvscw: 3 , the value of nivcsw: 0

[15116.188337] Another iteration of the loop has started,the value of nvscw: 4 , the value of nivcsw: 0

[15118.288338] Another iteration of the loop has started,the value of nvscw: 5 , the value of nivcsw: 0

(when interval is 3.1)

[15220.157728] simple module initialized

[15220.157743] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0

[15223.257748] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0

[15226.357768] Another iteration of the loop has started,the value of nvscw: 3 , the value of nivcsw: 0

[15229.457749] Another iteration of the loop has started,the value of nvscw: 4 , the value of nivcsw: 0

[15232.557745] Another iteration of the loop has started,the value of nvscw: 5 , the value of nivcsw: 0

(1) when interval is 0.1, 1.1, 2.1, 3.1, we could tell that interval between two iteration is equal to the interval we set. At this situation the value of nvscw is increase 1 when , the value of nivcsw: 0.

pi@raspberrypi:~ $ sudo insmod time\_fired\_module.ko log\_sec=1 log\_nsec=100000000^C

pi@raspberrypi:~ $ tail -f /var/log/syslogOct 7 17:54:11 raspberrypi kernel: [10446.978553] Another iteration of the loop has started,the value of nvscw: 5 , the value of nivcsw: 1

Oct 7 17:54:13 raspberrypi kernel: [10446.978553] Another iteration of the loop has started,the value of nvscw: 6 , the value of nivcsw: 1

Oct 7 17:54:13 raspberrypi kernel: [10448.078550]

Oct 7 17:54:13 raspberrypi kernel: [10449.178553] Another iteration of the loop has started,the value of nvscw: 7 , the value of nivcsw: 1

Oct 7 17:54:16 raspberrypi kernel: [10449.178553] Another iteration of the loop has started,the value of nvscw: 8 , the value of nivcsw: 1

Oct 7 17:54:16 raspberrypi kernel: [10450.278550]

Oct 7 17:54:16 raspberrypi kernel: [10451.378549] Another iteration of the loop has started,the value of nvscw: 9 , the value of nivcsw: 1

Oct 7 17:54:18 raspberrypi kernel: [10451.378549] Another iteration of the loop has started,the value of nvscw: 10 , the value of nivcsw: 1

(2) in the above result, nvscw would increase one when there is a new iteration. Nivcsw stays the same and is equal to one. Compare to each interval between two iteration, we could find that tho iteration execute in the same time. Because nivcsw indicates the number of events which share same CPU. So it is reasonable to two iterations share same CPU and execute in the same time.

**“Multithreading Design and Evaluation “**

*We modified the single thread program by*

* *Creating four pointers to four kernel threads.*
* *Modifying the module’s init to create and spawn four threads and also to bind the threads to separate cores by calling*

*kthread\_create(thread\_func,"data for thread\_func","mythread") four times and*

*kthread\_bind(&task\_pointer0,[0,1,2, or 3]) four times binding each thread to separate cores*

* *Waking up the processes four times so each process begins to run another iteration.*

1. *Similar to the single thread program, the iteration of each loop is the same as the specified interval(that is, the module parameter specifies the timeout value of the timer). However, in this case since we are running on four cores, the loop runs four times concurrently when the timer expires and the callback function is invoked.*

*(2) Since we are running on multiple cores the total number of context switches is lower than the context switch that happened when we were running on a single core because process threads are divided among the different processors.*

(when time interval is 0.1)

[16532.372539] simple module initialized  
[16532.472573] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:1   
[16532.472577] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:2   
[16532.472581] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:0   
[16532.472581] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:3   
[16532.572554] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:3   
[16532.572558] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:2   
[16532.572561] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:1   
[16532.572564] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:0

(when time interval is 1.1)

[16645.187961] simple module initialized

[16646.287989] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:1

[16646.287993] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:0

[16646.287998] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:2

[16647.387985] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:3

[16647.387990] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:1

[16647.387994] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:0

[16647.387998] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:2

[16648.487984] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:3

(when time interval is 2.1)

[16745.937738] simple module initialized

[16748.037771] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:1

[16748.037776] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:0

[16748.037780] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:2

[16750.137765] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:3

[16750.137770] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:0

[16750.137773] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:3

[16750.137778] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:1

[16752.237770] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:2

(when time interval is 3.1)

[16860.841268] simple module initialized

[16863.941300] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:0

[16863.941304] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:1

[16863.941308] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:2

[16867.041335] Another iteration of the loop has started,the value of nvscw: 1 , the value of nivcsw: 0 ,it's running on CPU:3

[16867.041340] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:1

[16867.041344] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:2

[16867.041348] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:0

[16870.141303] Another iteration of the loop has started,the value of nvscw: 2 , the value of nivcsw: 0 ,it's running on CPU:3

Multiple threads:(when nivcsw=1)

Oct 7 18:39:53 raspberrypi kernel: [13188.344606] Another iteration of the loop has started,the value of nvscw: 18 , the value of nivcsw: 0 ,it's running on COct 7 18:46:51 raspberrypi kernel: [13606.727987] Another iteration of the loop has started,the value of nvscw: 3 , the value of nivcsw: 1 ,it's running on CPU:2

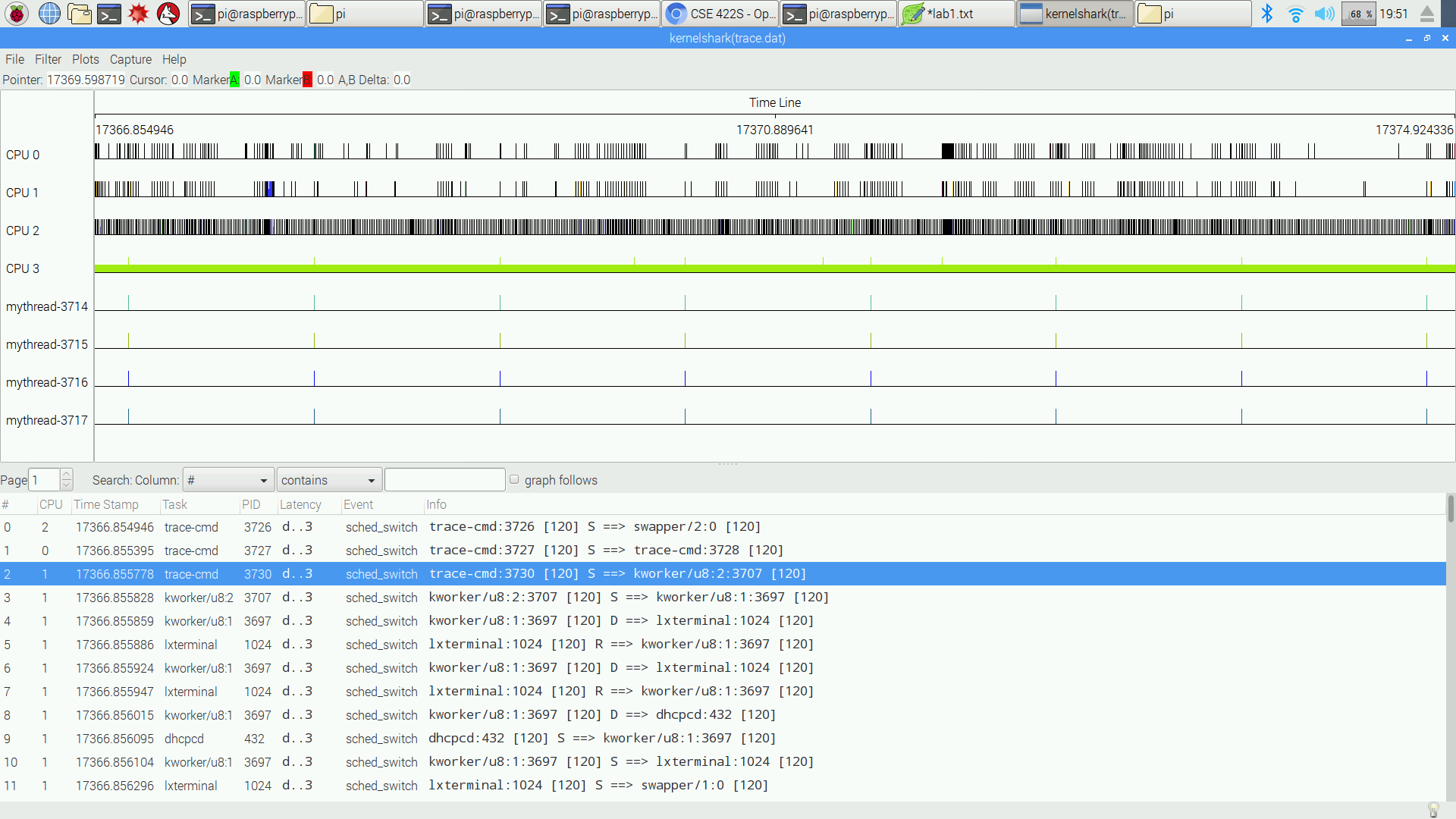
Oct 7 18:46:51 raspberrypi kernel: [13606.727991] Another iteration of the loop has started,the value of nvscw: 3 , the value of nivcsw: 1 ,it's running on CPU:1

Oct 7 18:46:54 raspberrypi kernel: [13609.727983] Another iteration of the loop has started,the value of nvscw: 3 , the value of nivcsw: 1 ,it's running on CPU:3

Oct 7 18:46:54 raspberrypi kernel: [13609.727988] Another iteration of the loop has started,the value of nvscw: 4 , the value of nivcsw: 1 ,it's running on CPU:1

Oct 7 18:46:54 raspberrypi kernel: [13609.727992] Another iteration of the loop has started,the value of nvscw: 4 , the value of nivcsw: 1 ,it's running on CPU:2

6.

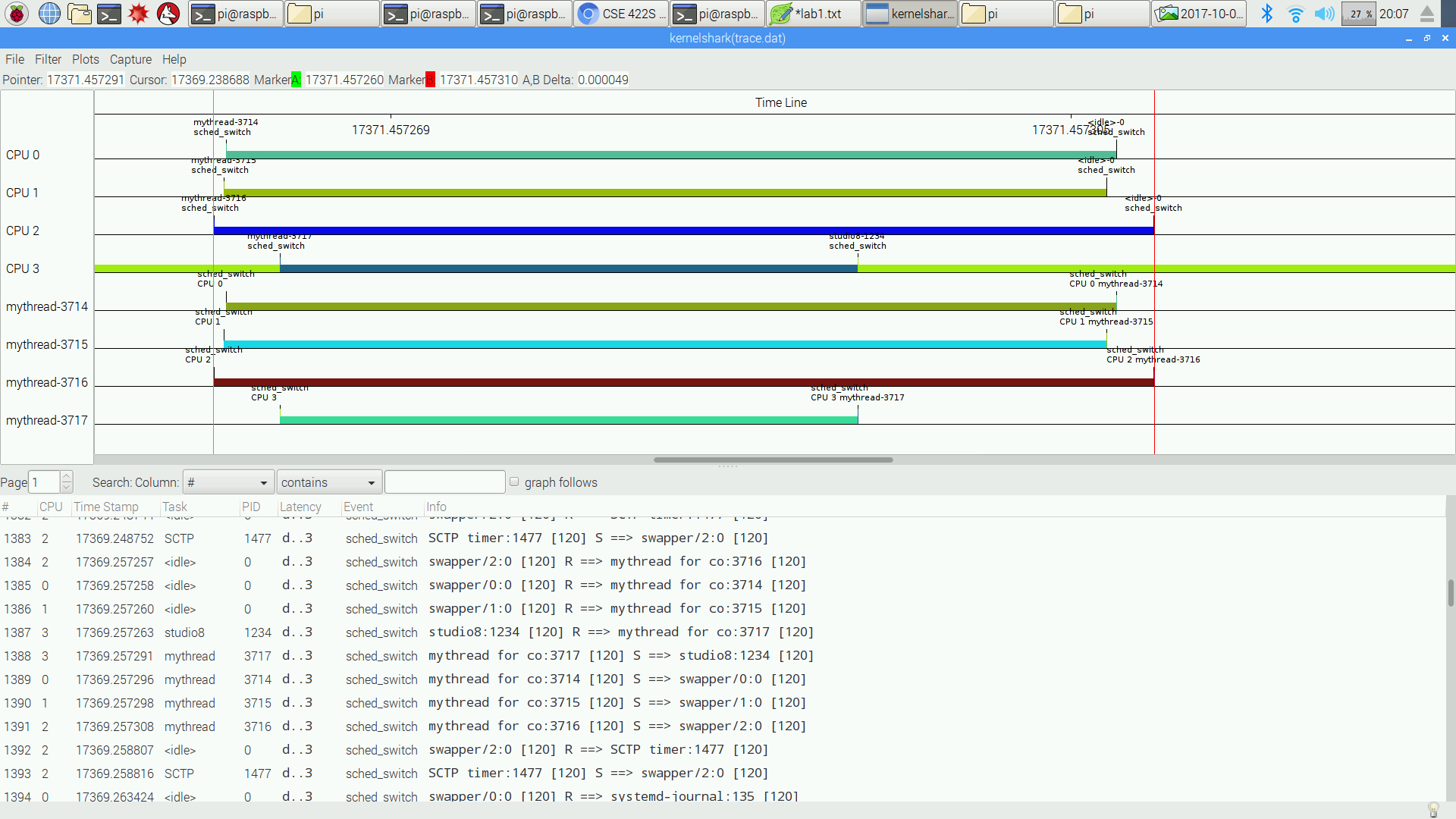


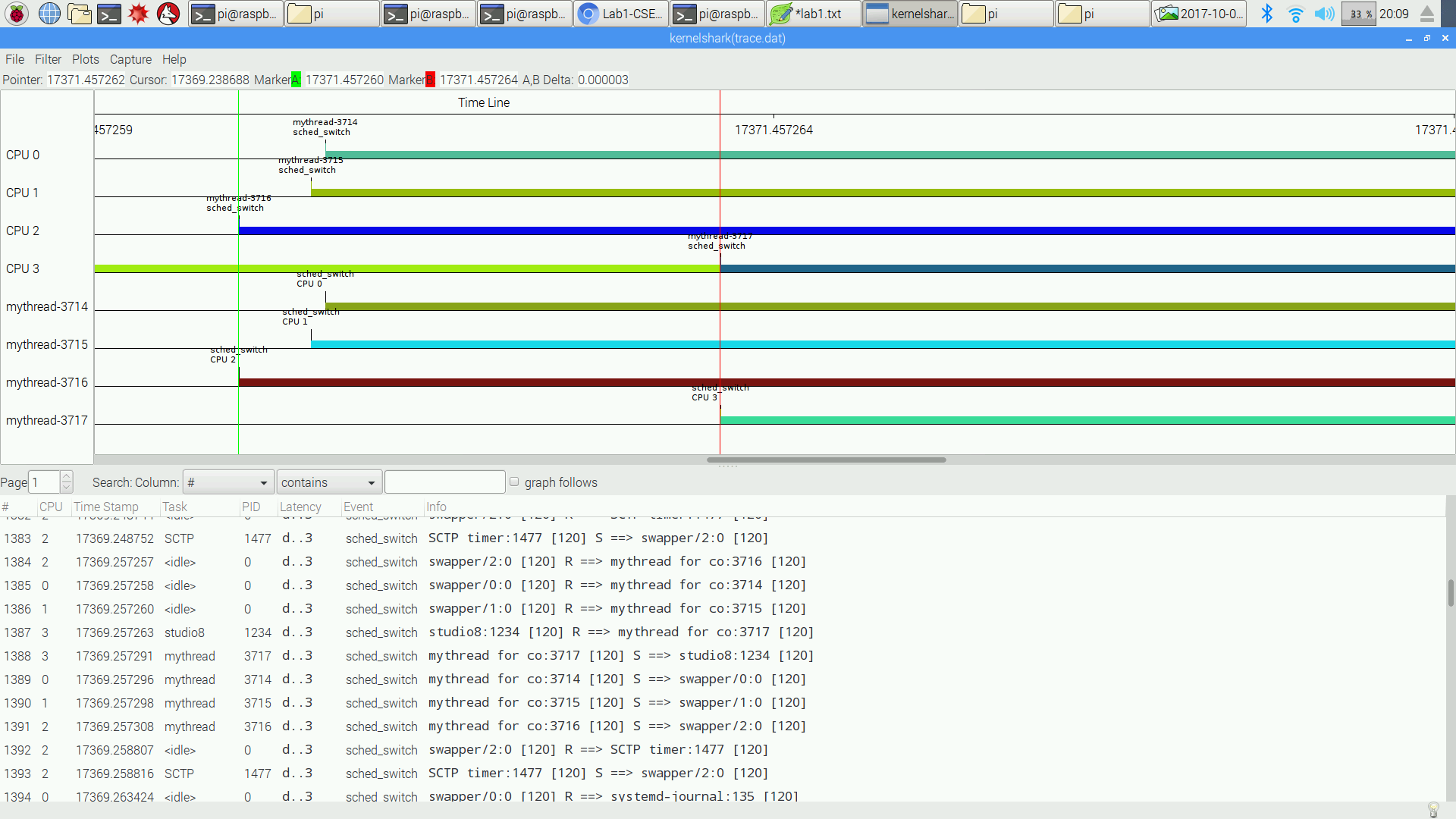
**"System Performance"**

(1) from the kernel, there is no preempted.

(2)rmmod and insmod can preempt kernel mode

(3)0.000049

****

(4) 0.000003  


(5) min: 0.000001, max:0.000003, avg: 0.0000016   
8.

6 hours