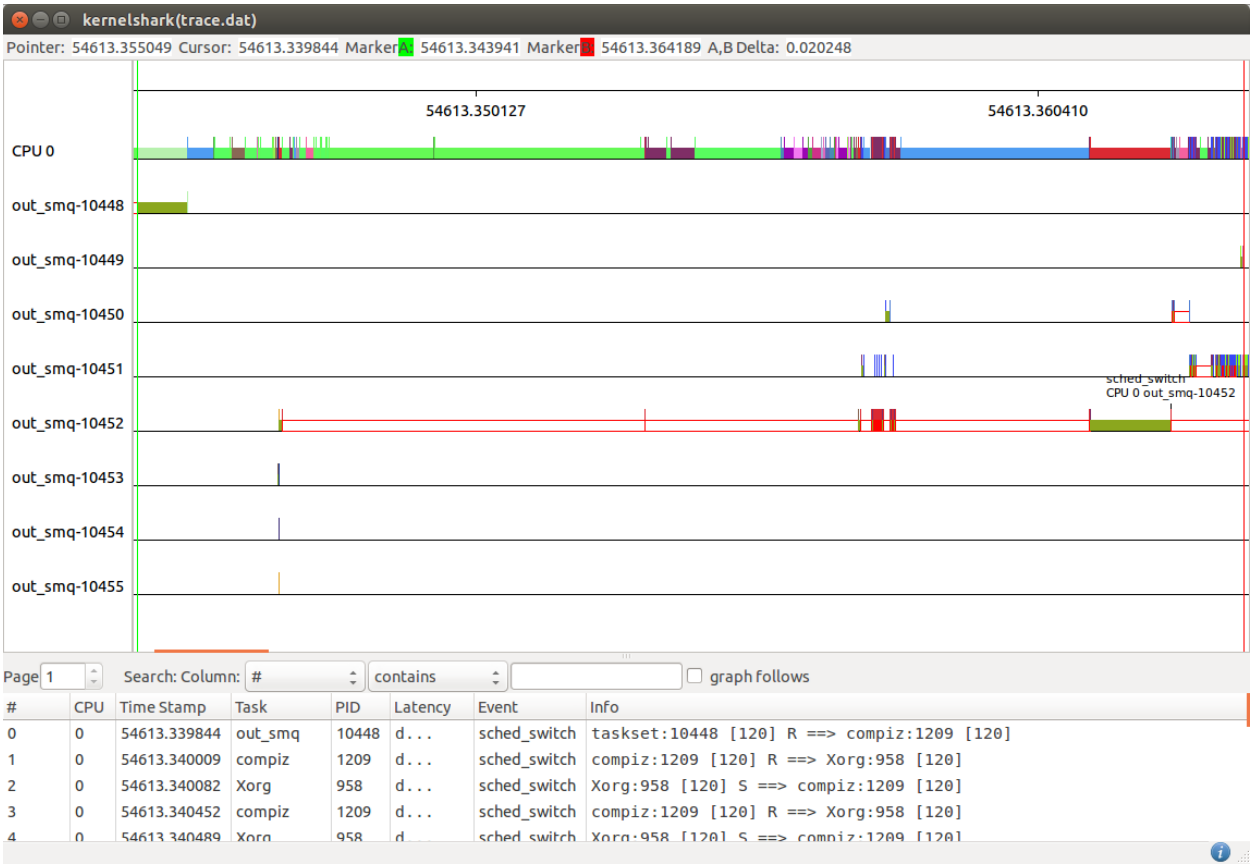
ESP Assignment 1 Part 1 Report

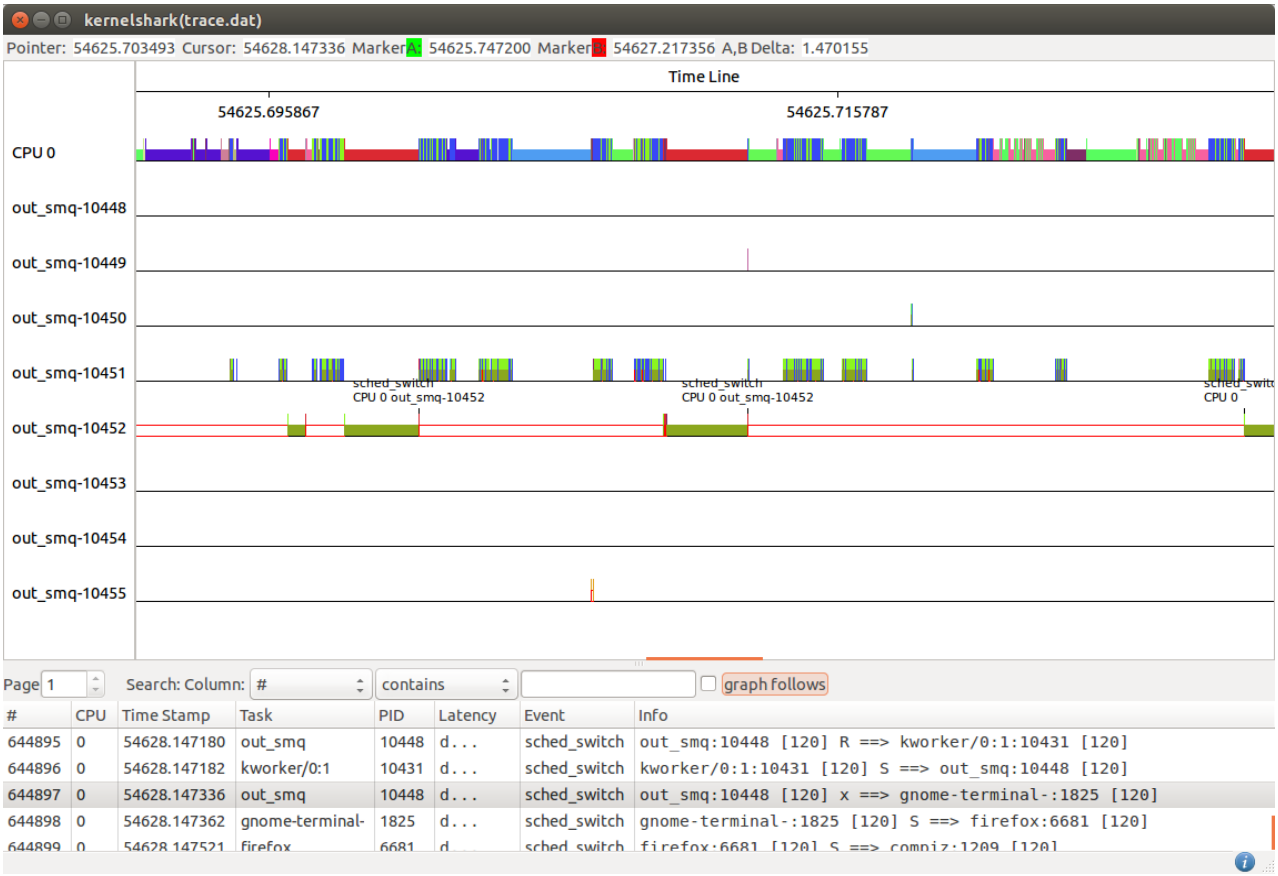
The output was executed on single core. Graph 1 represents the beginning of the main program and creation of threads. **Task name: out_smq**

Thread Information

Function	PID	Priority level
Main program	10448	
Periodic Thread 1	10449	81
Periodic Thread 2	10450	82
Periodic Thread 3	10451	83
Periodic Thread 4	10452	84
Aperiodic Thread 1	10453	70
Aperiodic Thread 2	10454	71
Receiver Thread	10455	91



Graph 1

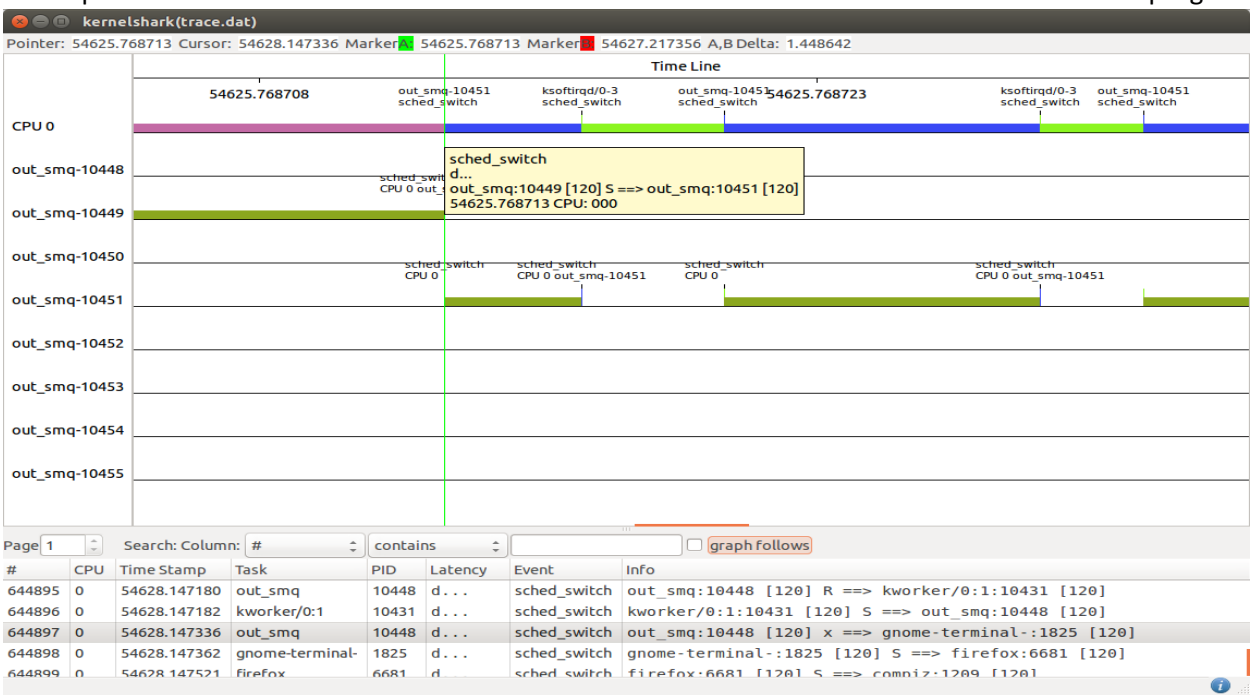


Graph 2

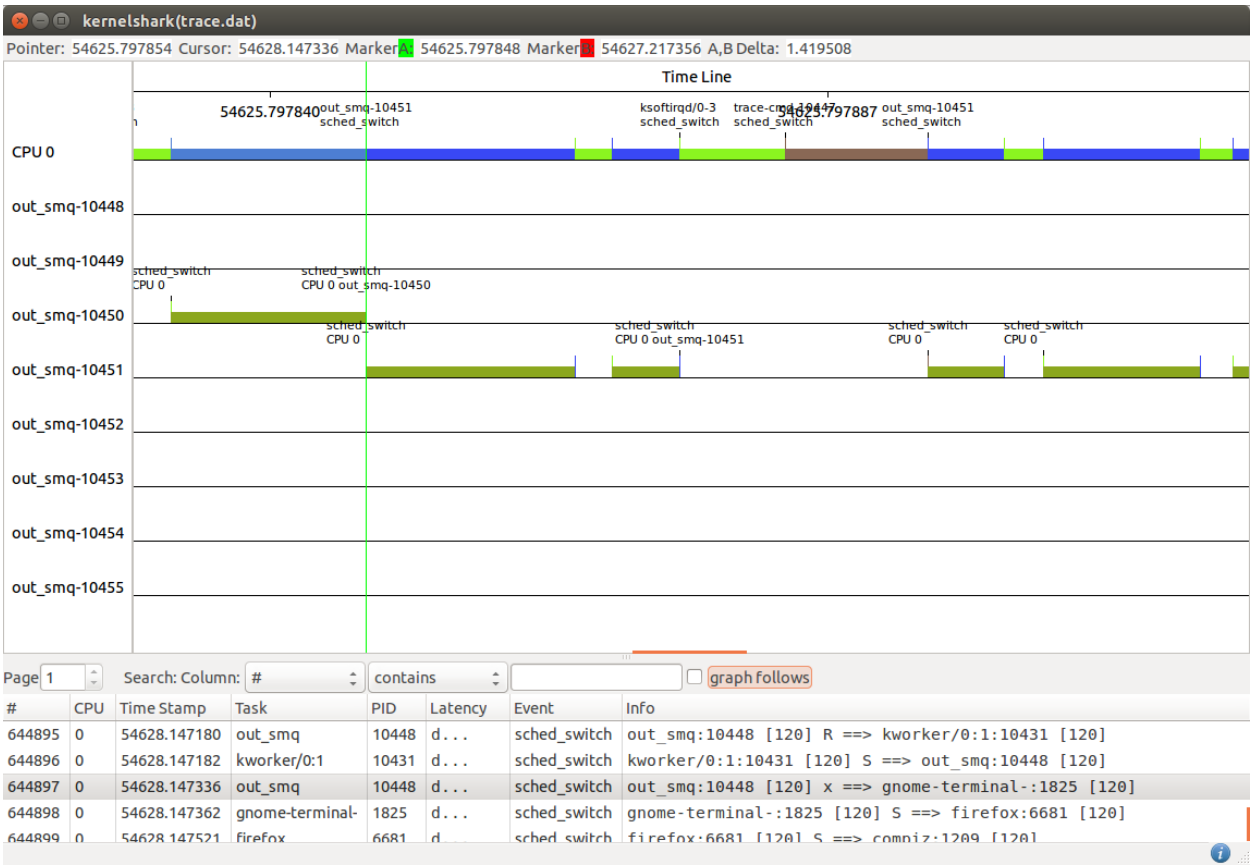
Graph 2 periodic threads running. From Graph 2, thread 10449 has the lowest priority and has less CPU time for enqueue operation and can be preempted by other tasks which has higher priority level.

During Enqueue operation periodic thread 10452 gets more CPU time and is not preempted by other periodic threads. However, thread 10452 can be preempted by the receiver thread 10455 which has the highest level of priority in the application.

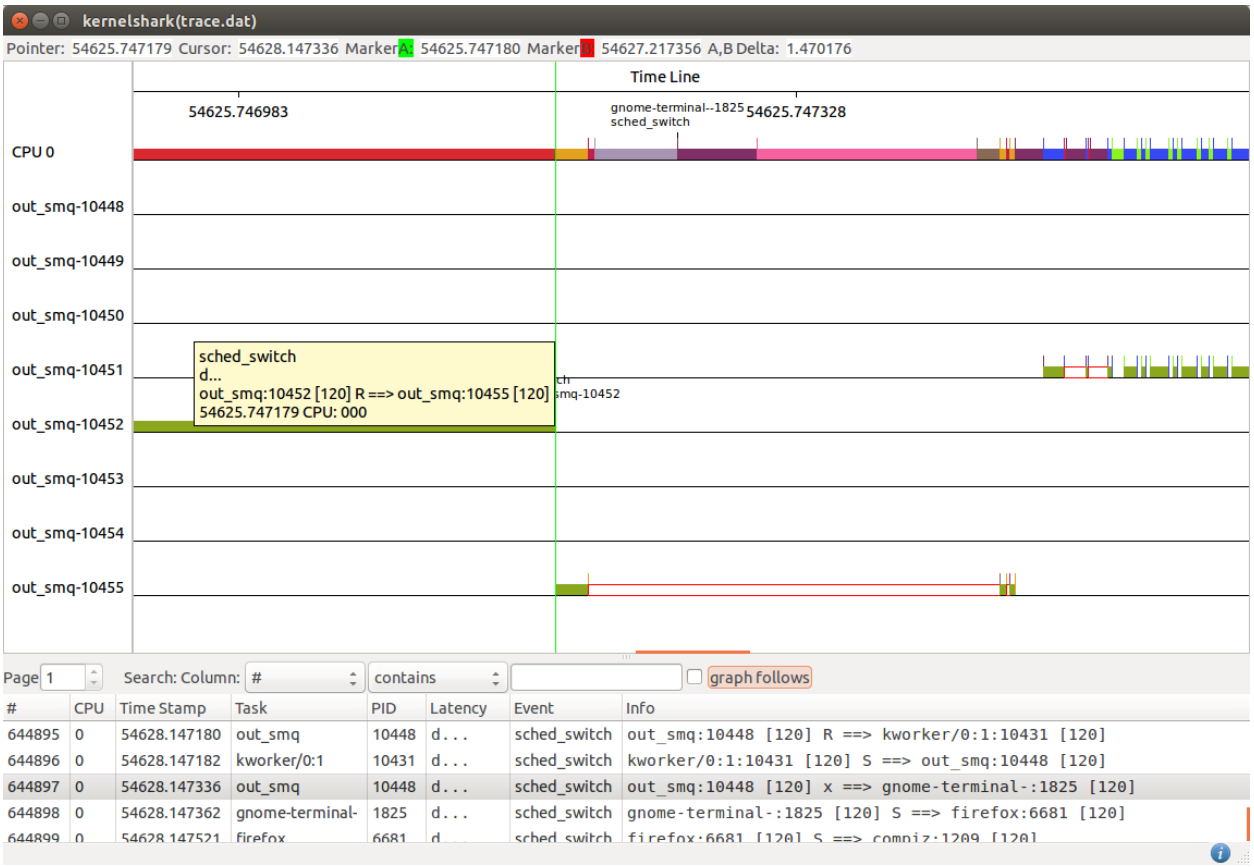
Preemption in the program



Graph 3



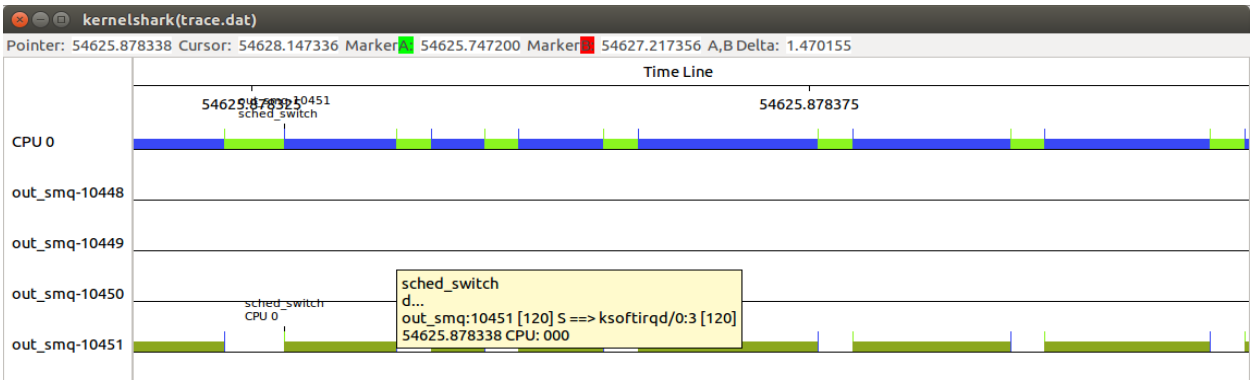
Graph 4



Graph 5

Graph 3, 4, 5 depicts schedule switching between events. In Graph 3 thread 3 takes over CPU from thread 1. In Graph 4 thread 3 takes over CPU from thread 4. In Graph 5 receiver thread takes over CPU from thread 4.

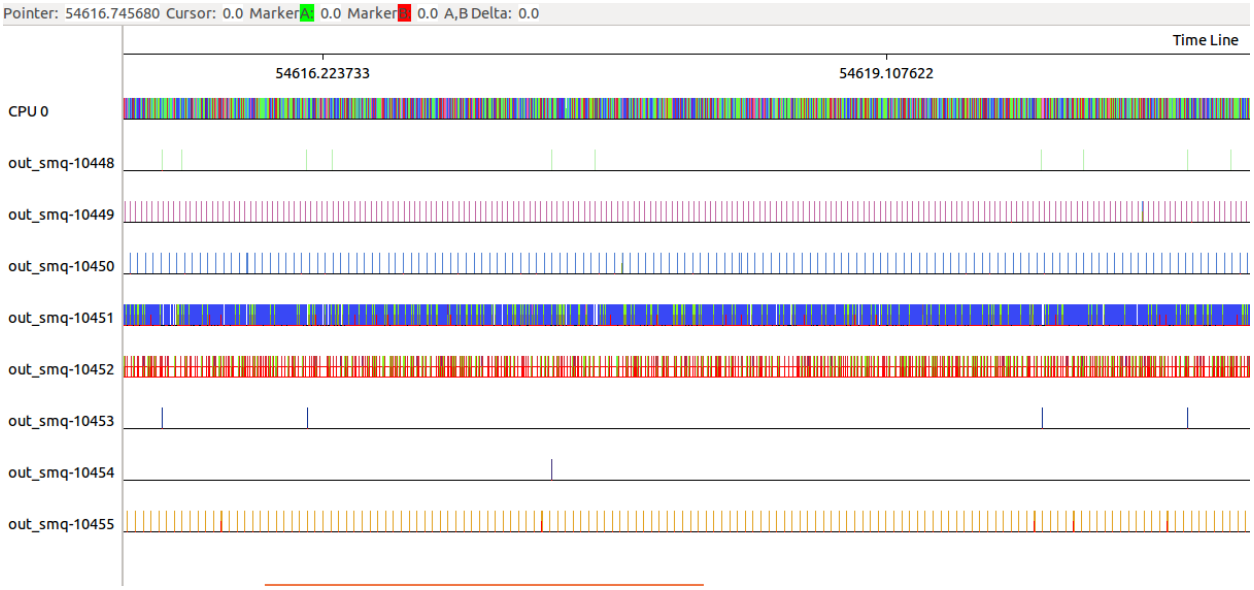
During the execution of threads, they are preempted by various other internal threads with high priority for example watchdog, interrupt request handler, etc.



Graph 6

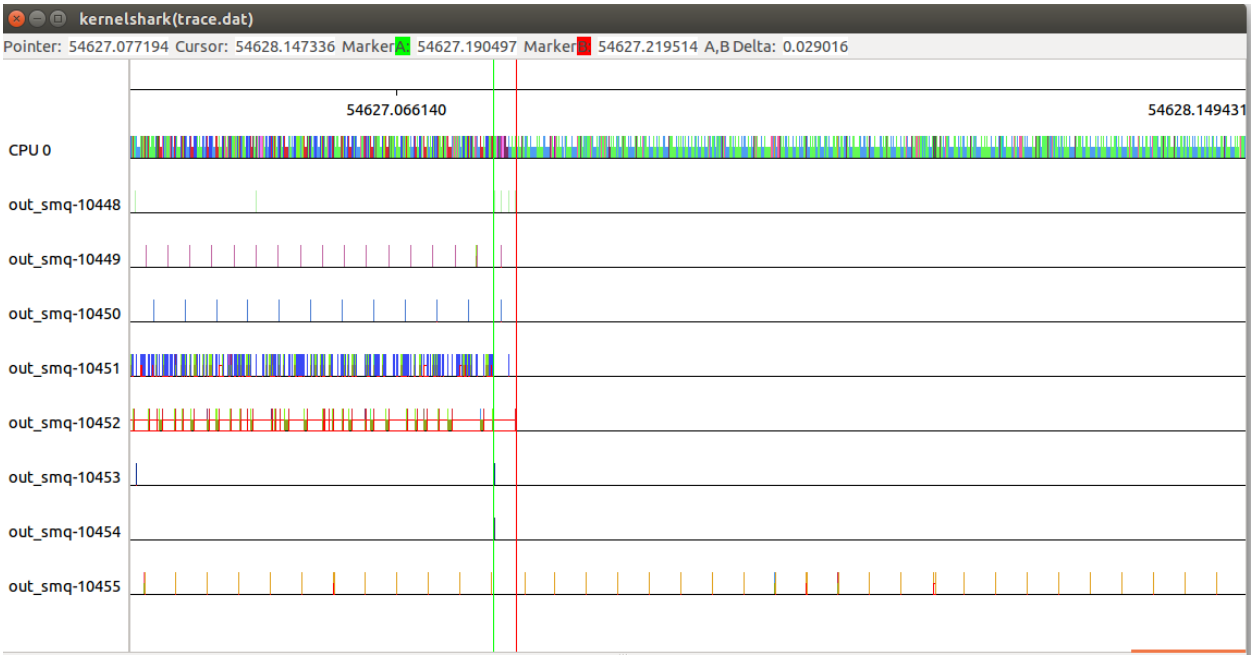
In Graph 6 CPU 0 is switched between thread 3 and ksoftirqd(software interrupt queue) thread 3 cannot have the CPU until the task is done as ksoftirq has very high priority. Since the program is running on single

core the thread cannot be pushed to different CPU, the thread execution is paused until the ksoftirqd schedule switches.



Graph 7

Graph 7 lists total duration of the threads running. Aperiodic thread 10453 and 10454 events are generated only during mouse click.



Graph 8

Graph 8 represents program termination after double click mouse event. In the Graph 8 double click is detected from thread 10453 and all the threads (4 periodic and 2 aperiodic) are terminated (between Marker A and B duration). However main program waits for receiver thread to finish. Thus, the receiver thread is active until all elements are dequeued and then terminated.