```
// The simulation running in a thread
function [sim, outlist, userdata]=Thread_MainRT(sim, inlist, userdata)
  [sim, Tpause] = ld_const(sim, 0, 1/27);  // The sampling time that is constant at 27 Hz
  [sim, out] = ld_ClockSync(sim, 0, in=Tpause); // synchronise this simulation

  // print the time interval
  [sim] = ld_printf(sim, 0, Tpause, "Time interval [s]", 1);

  // save the absolute time into a file
  [sim, time] = ld_clock(sim, 0);
  [sim] = ld_savefile(sim, 0, fname="AbsoluteTime.dat", source=time, vlen=1);

  outlist = list();
endfunction

// Start a thread
ThreadPrioStruct.prio1=ORTD.ORTD_RT_NORMALTASK; // or  ORTD.ORTD_RT_REALTIMETASK
ThreadPrioStruct.prio2=0;  // for ORTD.ORTD_RT_REALTIMETASK: 1-99 (man sched_setscheduler)
                           // for ORTD.ORTD_RT_NORMALTASK this is the unix nice-value
ThreadPrioStruct.cpu = -1; // The CPU on which the thread will run; -1 dynamically assigns to a CPU,
                           // counting of the CPUs starts at 0

[sim, StartThread] = ld_initimpuls(sim, 0); // triggers the computation only once
[sim, outlist, computation_finished] = ld_async_simulation(sim, 0, ...
                             inlist=list(), ...
                             insizes=[], outsizes=[], ...
                             intypes=[], outtypes=[], ...
                             nested_fn = Thread_MainRT, ...
                             TriggerSignal=StartThread, name="MainRealtimeThread", ...
                             ThreadPrioStruct, userdata=list() );
```