Exercise A - Real-time Architectures

A given system is connected to its environment via an input and an output port. Communications via both ports have stringent timing requirements.

- The input is provided with a rate of 50 times per second; it has to be retrieved within 10 msec. Accepting an input and copying it into an internal variable takes 6 msec. This is represented in pseudo code as an assignment of the form *x := InPort*;
- The output must be provided with a rate of 25 times per second. It has to be unchanged for at least 10 msec. It takes 5 msec. to copy an internal variable to the output port.

The environment is responsible for delivering the input with the specified rate. Each output is computed based on two inputs. It takes 21 msec. to perform such a computation.

1. Identify the real-time tasks in this system and express them in pseudo code. (Depending on your point of view there can be one or two tasks.) Discuss the schedulability with RMS and EDF. (Utilization bound, determine response times, investigate the phasing).
2. It is necessary to decompose the tasks such as to make the system schedulable. Introduce variables for communication between the tasks; explain how many are needed. This introduces dependencies of a producer-consumer nature. Take care of correctness, either through explicit synchronization or through scheduling. Repeat the analysis. Are all tasks periodic?

An improvement of this system uses boolean variables associated with both ports, viz., *Bi* for the input port and *Bo* for the output port. A change in value indicates that a new value has been written to the port. All outputs must now be based on genuine inputs.

3. Explain how jitter at the input affects the original system and this improved system. What is the effect of an absolute jitter of max. 50% (note: absolute jitter denotes the maximum deviation from the specified arrival time on a per event basis). Introduce sufficient buffering in the system to deal with this. Notice that we do want to generate the output with the correct rate. Repeat the schedulability analysis.
4. Assume that we implement this system by a direct mapping of tasks to threads. Shared variables (including the buffers) are used under exclusion.
    a. Can there be (unbounded) priority inversion?
    b. Discuss which concurrency control policy can be used; explain if there are differences in the current example.
    c. Thread switching takes time and usually is time-driven. Suppose that a switch takes 0.5 msec. and that the corresponding clock resolution is 50 hertz (i.e., switching can only happen at 20 msec. boundaries).
        i. Include this switching time in the reference model. Is it safe to just add it to the computation time?
        ii. How does this affect the current system? If this 50Hz is not fast enough, determine a switching time that does work.

Make a clear and concise report of this exercise and hand it in on paper. In cases you cannot find a complete calculation, describe the issues at least qualitatively.