# Exercise session 5: software architectures (part 3)

Thomas Braipson

November 6, 2026

# The RTOS architecture

A few tips:

- Identify which tasks can/must be implemented inside interrupt routines (same criteria as previously)
- Identify periodic and non-periodic (non-urgent) tasks
- Assign a priority to each task
- Identify communication objects between tasks (semaphores, message queues , . . . )
- Do not use OS services inside interrupt routines if:
  - The service is blocking
  - The service may cause a context switch and the OS is not aware that the program currently is in an interrupt routine

# Exercise 1: Implement a solution to the following problem

3. [*6 points*] The onboard microcontroller of a homemade quadcopter has to receive and execute instructions sent by the pilot via a wireless link (such as taking off, landing, flying in a specified direction, ...). In addition to processing those instructions, it constantly stabilizes the vehicle. In order to monitor and control stability, the quadcopter is equipped with two 3-axis accelerometers and one gyroscope. Moreover, it includes a camera whose signal, along with others monitoring data, has to be sent back to the pilot. The microcontroller has to perform the following tasks:

   - receiving and processing instructions: 10 times per second, execution time = 5 ms;
   - controlling the motors: 100 times per second, execution time = 2 ms;
   - acquiring data from sensors: period of 4 ms, execution time = 1 ms;
   - sending the video signal and other data: period of 40 ms, execution time = 20 ms.

   (a) What is the best software architecture for this system? Justify.

# Exercise 2: Implement a solution to the following problem

3. A nanosatellite in charge of taking pictures of the earth is controlled by an embedded system. This system is equipped with a radio transceiver that receives telecommand data packets from a ground station. Upon receiving a data packet, the radio transceiver sends an interrupt request to the onboard microcontroller. Such requests are always separated by a delay of at least one second.

   The microcontroller has to perform the following tasks:

   - $\tau_1$ fetches and processes the data packets communicated by the radio transceiver;
   - $\tau_2$ performs computations for estimating the position of the satellite, every 10 ms;
   - $\tau_3$ regulates a DC/DC power converter, every 2 ms;
   - $\tau_4$ acquires an image from a camera, after telecommand data requesting to perform this operation has been received from the ground station. It then performs image processing operations.

   Tasks $\tau_1$ and $\tau_3$ execute in less than 0.1 ms; task $\tau_2$ needs 1 ms. The execution of task $\tau_4$ may require up to 500 ms of CPU time, depending on the image processing operations that must be carried out.

   (a) What is the best software architecture for this system? Justify. [3/20]