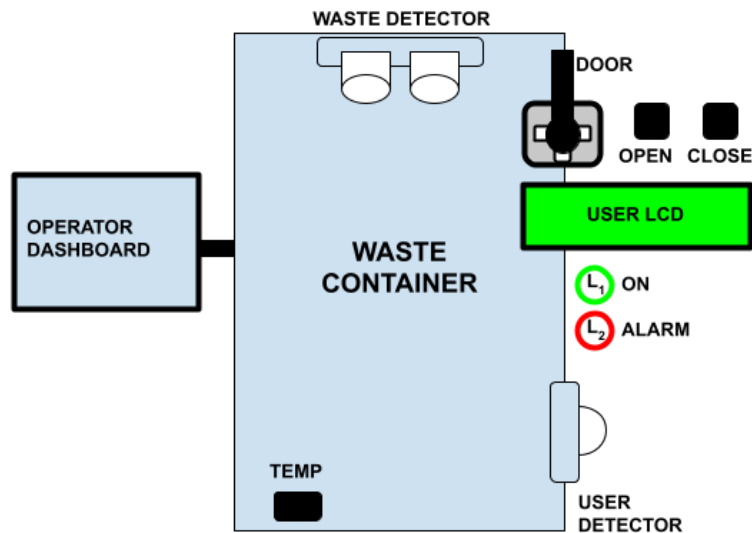


## Assignment #2 - *Smart Waste Disposal System*

v1.0.0 - 20241031

We want to realise an embedded system called *Smart Waste Disposal Container*. The prototype is meant to implement a simple smart system for waste disposal (dangerous liquid waste in particular).



Hardware components:

- **USER DETECTOR** is a PIR, to detect the presence of a user in front of the container
- **WASTE DETECTOR** is a sonar, to measure how much the container is full
- **DOOR** is a servo-motor, controlling a small door, to enter waste
- **USER LCD** is a I2C LCD, used to interact with users
- **OPEN** and **CLOSE** are two tactile buttons
- **L1** is a green led, to signal when the system is available working normally, **L2** is a red led, to signal when the system is not available.
- **TEMP** is an analog temperature sensor

The embedded subsystem is meant to communicate by means of the serial line with a PC, running a program called **OPERATOR DASHBOARD** (for operators). In particular the **OPERATOR DASHBOARD** is a GUI application providing features to:

- track and monitor the state of the container
  - level of waste
  - temperature
- a **EMPTY** button to empty the container

- a RESTORE button to restore the container (in the case of alarm)

## Description

Basic behaviour: a user can dispose of some waste (liquid in particular) by pressing the OPEN button. This opens the DOOR, so that the user can enter (spill) the waste, pressing the CLOSE button to close the DOOR. When/if the container becomes full, it is no longer available for accepting waste and it waits for being emptied by operators.

### Detailed behaviour

- At the beginning, when turned on, the container is meant to be empty and available
  - Led L1 is on, signaling availability
  - DOOR is closed (the motor is in position 0 degree)
  - USER LCD displays the message: PRESS OPEN TO ENTER WASTE
- If no user is around for some amount of time  $T_{sleep}$ , the system goes to sleep
  - only Led L1 is kept on (or Led L2, when it is already on)
  - the system awakes when a user is detected by the USER DETECTOR in front of the container
- If/When the user presses OPEN, the DOOR is opened (by rotating the motor of  $+90^\circ$  degrees) and the user can spill the waste
  - in this stage, the USER LCD displays "PRESS CLOSE WHEN DONE"
  - the user has a maximum time  $T_1$  to enter the waste, after that the DOOR is automatically closed
- When the user presses CLOSE (or after the timeout), the DOOR is closed (by turning back the motor to  $0^\circ$  degrees) and a message "WASTE RECEIVED" is displayed for  $T_2$  seconds, before going back to accepting waste state
- By entering waste, the container can become full. This is detected by the WASTE DETECTOR, by measuring the "waste level" (from the distance), that is: If the distance goes below some threshold  $D_1$ , then the container is considered full and in this case it is no longer possible to accept new waste until the container is emptied by operators
  - in this case, the USER LCD displays the message "CONTAINER FULL", the Led L1 is turned off and the Led L2 is turned on
  - If the container becomes full while a user is entering waste, then the DOOR is automatically closed
- A container that is full can be emptied by operators, acting on the OPERATOR DASHBOARD. In particular, an operator by pressing the GUI button "Empty the container" provided by the dashboard causes the DOOR to be opened on the reverse side (motor rotated to  $-90^\circ$  degrees) for  $T_3$  seconds, simulating the emptying process, and then it is closed back. After that, the Container goes back to its normal state
  - Led L1 is turned on, Led L2 is turned off and the message on the USER LCD is changed

- Operators can use the GUI Dashboard to inspect the current state of the container, in particular:
  - filling percentage (0% is empty, 100% full)
  - [OPTIONAL] the history
- The temperature sensor TEMP is used to continuously monitor the temperature inside the container. If the temperature exceeds some threshold MAXTEMP for more than MAXEMPTIME time interval, then the container does not accept anymore waste until the intervention of operators
  - in this case, the USER LCD displays the message "PROBLEM DETECTED", the Led L1 is turned off and the Led L2 is turned on
  - if a user was entering waste, the process is interrupted – the DOOR is closed
  - Operators are meant to restore the normal state of the container by pressing the RESTORE button on the Dashboard GUI

---

Develop the embedded software on Arduino + PC connected through the serial line, implementing the Arduino part in C++/Wiring and the PC part in Java or in another favourite language. The Arduino program must be designed and implemented using task-based architectures and synchronous Finite State Machines.

For any aspect not specified, you are free to choose the approach you consider more useful or valuable.

## The Deliverable

The deliverable consists in a zipped folder **assignment-02.zip** including two subfolders:

- **src** folder, including
  - two further folders: `arduino` and `pc`, the former must contain the Arduino project called `smart_waste_disposal`, and the latter the sources of the Java (or whatever language/platform) program
- **doc** folder, including:
  - a brief report describing the solution, in particular the diagram of the FSMs and the representation of the schema/breadboard – using tools such as TinkerCad or Fritzing
  - a short video (or the link to a video on the cloud) demonstrating the system