

Reactive System Programming

Esterel (Part 2)

SF7 (EPU-SI) / E2 (Master STIC)

November 6, 2006

In this Lab. you have to program simple reactive applications in the ESTEREL language.

Several ESTEREL files and documents are available at [//www.i3s.unice.fr/~andre/CADoc](http://www.i3s.unice.fr/~andre/CADoc) (direct access to this directory). Copy `td4St.zip` and extract files.

1 Clock (Continued)

A solution to question 2.3 in Lab.1 is available in `TD4\Programmes`.

1.1 Explore the solution

- Double-click on `TD4\Programmes\DisplayTime\DisplayTime.etp`
- Analyze ESTEREL files
- Record a scenario (Hints: assume a minute takes 6 seconds, an hour 6 minutes and a day 2 hours)

1.2 Improved solution

Add a pure input signal `WhatTimeIsIt` and an output signal `ItIs` whose type is `string`. Whenever `WhatTimeIsIt` is present, `ItIs` will be emitted with the current time written according to the following syntax (terminals are in bold font, non terminals in italic font): *hh:mm:ss* (**am|pm**)

To solve this problem declare an ESTEREL function:

```
function toTime(unsigned , unsigned , unsigned) : string ;  
% args : hours , minutes , seconds
```

and implement it in C.

Remark : the ESTEREL predefined type `string` is defined in C as follows:

```
typedef char * string ;
```

Moreover, containers for string values are declared as arrays of characters:

```
#define STRLEN 81  
...  
char foo_Vxx [STRLEN];
```

Question : write the C implementation of the toTime function.

1. Create two files DisplayTime.h and DisplayTime_data.c in TD4\Programmes\DisplayTime\Default\Simulation.
2. Add these files to the project in the Extern folder (see Fig. 1).
3. Simulate.

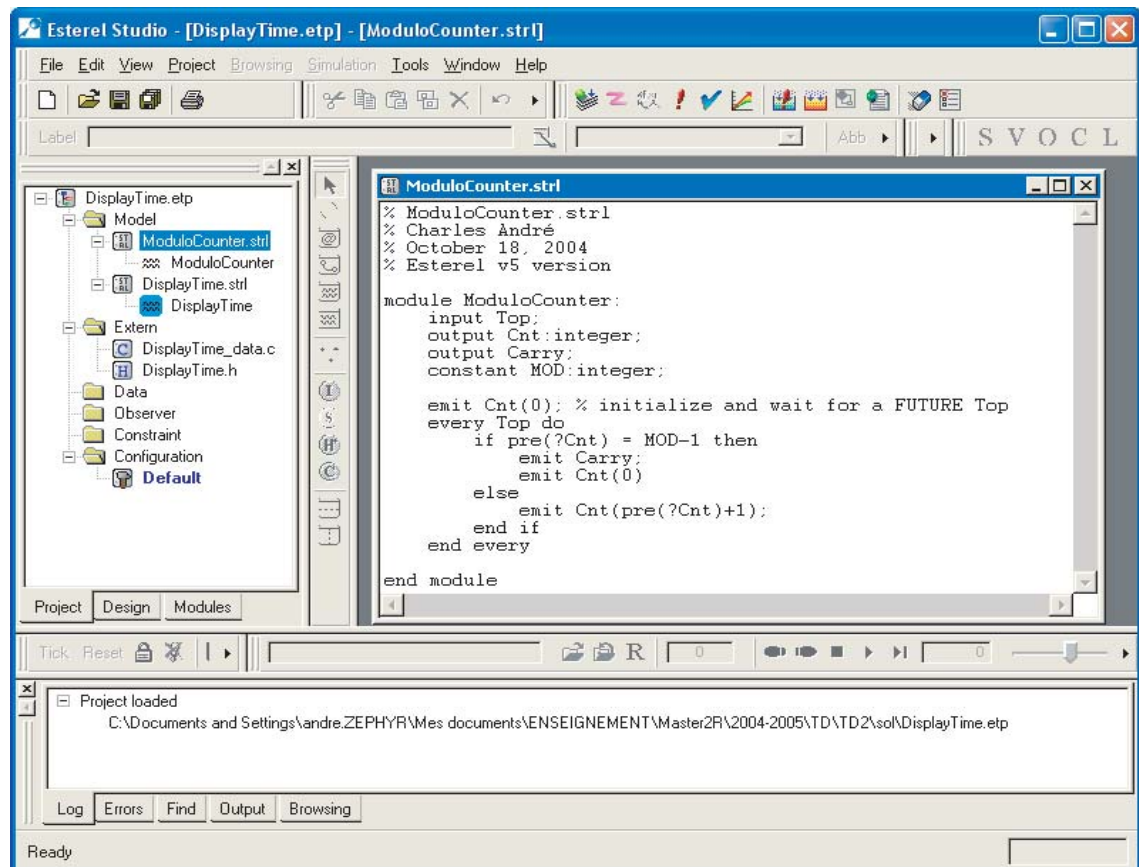


Figure 1: DisplayTime project.

2 Monostable

2.1 Non-retriggerable Monostable

Its interface and behavior are given in Fig. 2

The following program expresses this behavior:

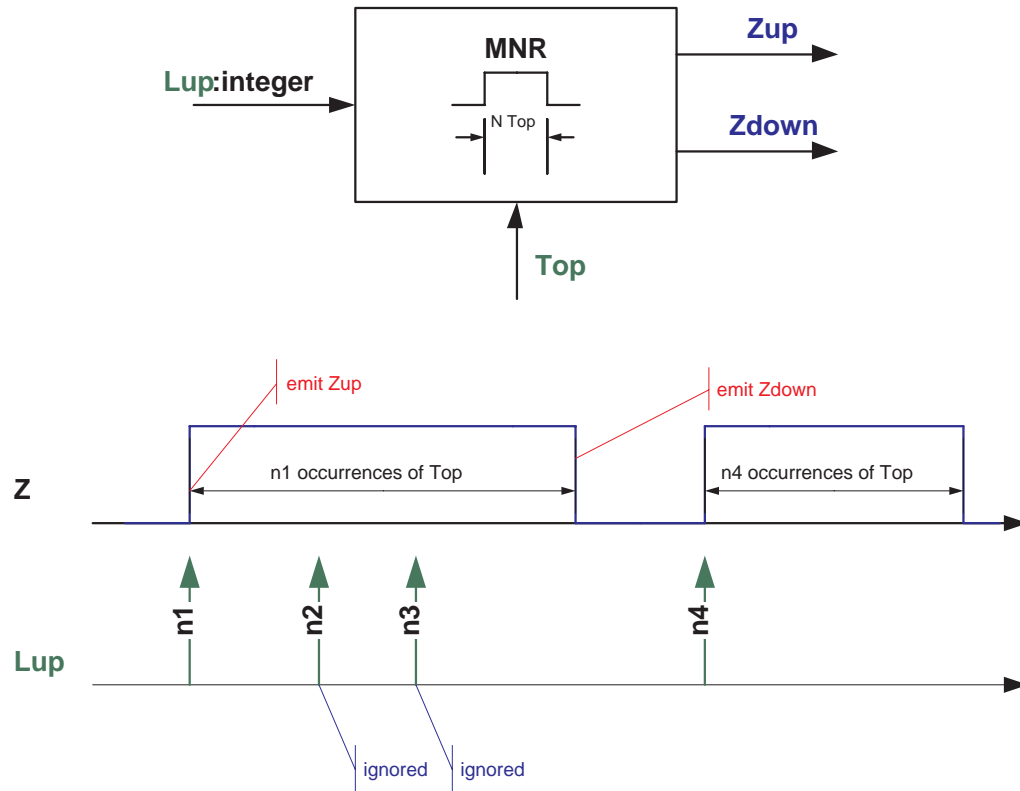


Figure 2: Non-retriggerable Monostable.

```
% Non-Retriggerable Monostable
% Charles André
% October 18, 2004
```

```
module NRM:
  input Lup:integer , Top;
  output Zup, Zdown;

  loop
    await Lup;
    emit Zup;
    await ?Lup Top;
    emit Zdown
  end loop
end module
```

Compile and simulate this program (use the scenario recorder and the waveform recorder).

2.2 Retriggerable Monostable

Change the previous code to a *retriggerable* monostable.

2.2.1 First approximation

Insert a temporal loop. It makes no matter whether **Zup** is emitted when the monostable is re-triggered.

2.2.2 Improved solution

Now, Zup must not be emitted when retriggering.

3 Level regulation in a tank

The level of a liquid in a tank must be kept between two levels L (low) and H (high). The tank is fed by a solenoid valve (V) and emptied by another one (E).

“Your mission, should you choose to accept it, is to control the level, without overflowing or underflowing. As usual, should you or any member of your team be drowned in the tank, the teacher will disavow any knowledge of your existence. This tape will self-destruct in five seconds. Good luck.”

We apply an all (valve wide-open) or nothing (valve closed) form of control, also known as a bang-bang control (Figure 3). The level sensors are *pressostats* (special kinds of manometers with a two-level output).

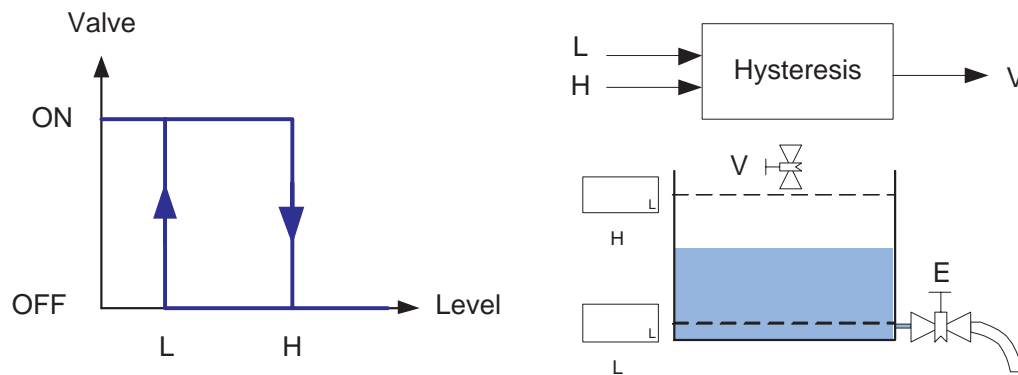


Figure 3: The system and the hysteresis control law.

3.1 Controller

Assume that the tank is initially empty.

Since input signals (coming from the pressostats) are two-valued (or Boolean) signals, they can be considered as *pure* ESTEREL signals, with the convention that the *present* status stands for a `true` (or high) value.

3.1.1 Event-driven solution

Assume that a level sensor generates events when the liquid level is *exactly* at its height. Propose and test an ESTEREL module to control the level.

3.1.2 A more realistic solution

A pressostat reads `true` when it is submerged and `false` when emerged. Take this remark into account for a new version of the controller. Use an ESTEREL relation for expressing the fact that sensor L is *below* sensor H.

Compile and simulate. What happens if the relation is violated?

3.1.3 Environment modeling

In this exercise, we model the behavior of the tank (and its contents) by an ESTEREL program.

Geometric specifications :

- the tank is a cylinder whose section is 0.1 m^2 and height is 1 m;
- the level in the tank (l) is represented by an integer.
- the elevation of the low-level sensor is 12 cm;
- the elevation of the high-level sensor is 88 cm;
- the throughput of valve V is 10 l/s when open, and 0 l/s when closed (no leakage!), and for valve E 5 l/s when open.

The program is supposed to react every second (1 Hz sampling period).

Question a: Write and test this program.

Question b: Simulate the controller *and* its environment.

Question c: Take account of inertia of valves. Assume a 1 s pure delay.