**Pre-defined or built-in triggering conditions**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Description | Inputs | Outputs | Config. params |
| periodiclabdata | Sends data every *ts* miliseconds | *t*, *ts* | All readable vars | *ts* |
| Periodic send-on-delta | Sends data when the change of a certain value (*y*) is bigger than *d*. Condition is checked on a periodic basis. | *t*, *ts*, *ti*, *y(t)*, *y(last)*, *d* | Subset of readable variables | *ts*, *ti*, *d,* definition of *y*, readable vars to send |
| Self-triggering | Sends data when the change of a certain value (*y*) is bigger than *d*. Condition is checked based on a customizable rule. | *t*, *ts*, *ti*, *y(t)*, *y(last)*, *d,* ***rule*** | Subset of readable variables | *ts*, *ti*, *d,* definition of *y*, readable vars to send, ***rule*** |

In the send-on-delta and self-triggering strategies, *y* could be the error of a controlled variable, a readable variable, the change of a readable variable, a writable variable or the change of a writable variable.

The periodiclabdata event is a subcase of the periodic send-on-delta case, where the condition is always met.

The periodic send-on-delta event is a subcase of the self-triggering case, where the sampling times are stipulated following a periodic basis.

There must be a mechanism to allow both plant owners/experts and client users to configure the parameters (set their values, choose the input/output variables, etc.).

However, the self-triggering strategy is the only one that is “open”, in the sense that either the lab owners/experts and the client users should be able to define the rule for determining the next sampling time in which the triggering condition will be checked. In the other two cases, these actors only need to configure things (see previous paragraph).

Where:

* *t* – Time past since the client connected to the lab.
* *ts* – Sampling period.
* *ti* – The time in which the first sampling is done 🡪 could be reformulated in terms of the value of *y* (i.e., take the first sample when *y*=0).
* *y(t)* – The value of the entity (the error of a controlled variable, a readable variable, a writable variable), at time *t*, over which the triggering condition will be checked.
* *y(last)* - The value of the entity, in the last sampling time, over which the triggering condition will be checked.
* *d* – The threshold that determines when to fire the event.
* ***rule*** – A rule that determines when the next sampling time should be.