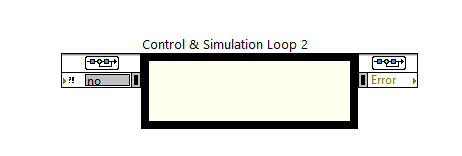
We have our system that contain a command part setpoint fixed by the user, a process part containing either a pump and a valve and a sensor who return the current value of the level of water, this system need to be regulated so we include PID regulator and a closed loop process.

Before doing anything if you want to modify and test your version just see those description it will really help you.

**To understand you have to try my values and don’t forget to press debut simulation**

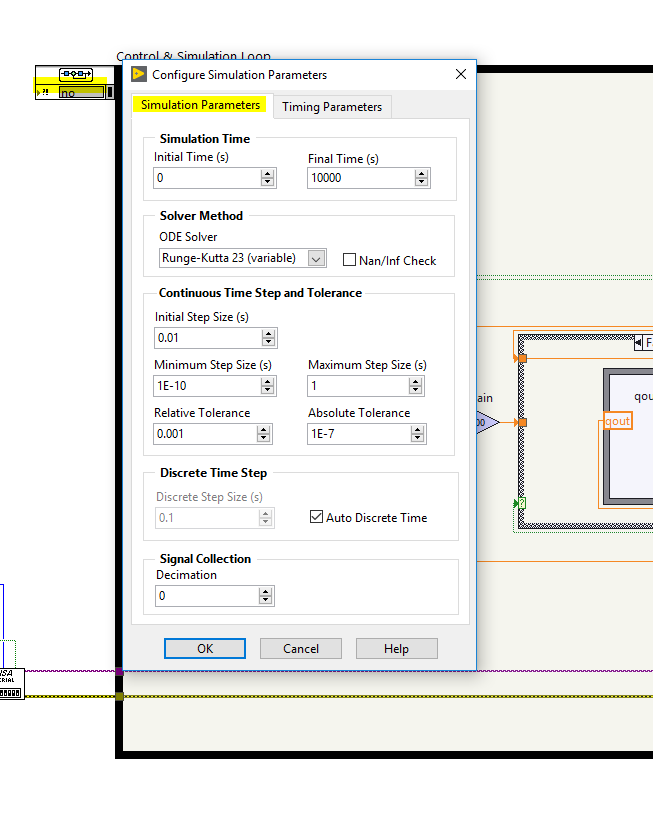
**First PART**

The first thing to do is to put the control and simulation loop (control & simulation->simulation-> control & simulation loop).



You can modify the processing time of this loop (in our case we putted 10000 as final time

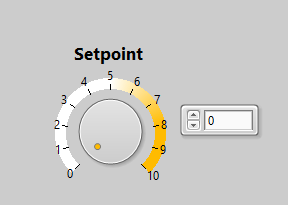
And we modify the timing parameters by choosing synchronize loop to timing source with a period of 50)



**Command Part:**

**Setpoint:**

Simple double control

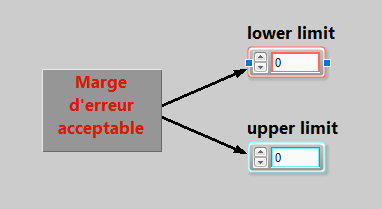


Front Panel Block diagram

**margin of error**

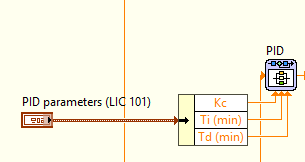
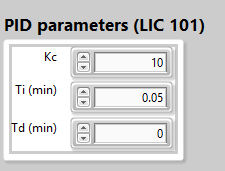
the acceptable values of output that you want to get.

If lower limit< setpoint – output < upper limit -> we stop the pump and we closed the valve



**Regulator PID:**

To find the PID you have to search in control & simulation-> simulation-> continuous linear systems



Front Panel Block diagram

In front panel: you can input any value of the parameters KC, Ti, Td (I putted the values that I found work fine for our system in the front panel) we use cluster to group all the variables in the same area.

In block diagram: we use unbundle by name and we wired to PID (we have to enter to PID and change to terminals all the variables)

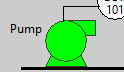
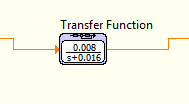
Process part:

**Pump:**

We present the pump by a transfer function of first degree 0.008/(0.016+s) (you can choose another function of transfer)

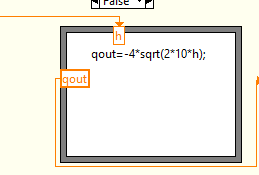
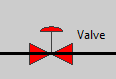
To find the transfer function search in control & simulation simulation-> continuous linear systems-> Transfer Function (double click and enter your parameters)

In front panel when the colour of pump is green means the pump is on otherwise red means off

Front Panel block diagram

**Valve:**



Front Panel Block diagram

The valve in our case have two state: fully opened or fully closed

so, the flow depends of the section and the level of water.

The input is h: level of water

qout: water flow

we use a formula node (in structure section)

to add an input, you have to click in the border of formula node and we can see there’s many choices (choose add input, add output) and name those variables.

**Sensors:**

We can use (ultrasonic or electrical sensors) in your model.

In our case we used a unitary return because the input and the output have the same type.

The functioning of our system:

We have three condition:

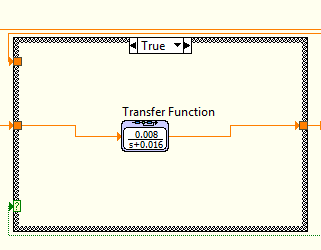
1:

If the setpoint > output -> we turn on the pump in order to fill the tank to the level what we want

Here the regulator PID insure that the regulation of the input we use only the proportional to speed up the procedure of filling and integrator to cancel the error.

We use a case function and the case selector we put the Boolean result of the simple minus operation between the input and the output

In our case True

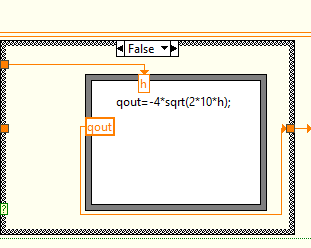


2:

If the setpoint<output

We turn on the valve (PID has the same function in the first condition)

so, the case is false

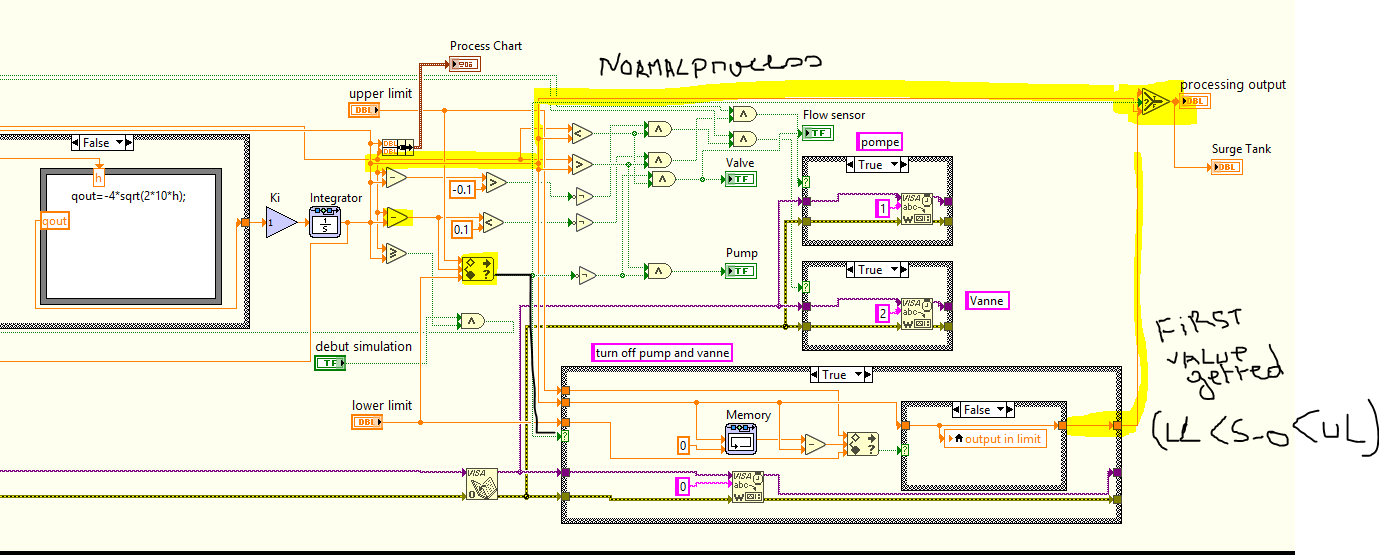


3:

If lower limit < setpoint – output <upper limit

We turn off the pump and we close the valve.

In our case we use case selector and in selector we put the Boolean result of the equation we written in this section and in the upper we put the first values we got when this inequation is valid (to see the output fix) and in the downer we putted the normal result coming either from the pump or the valve.

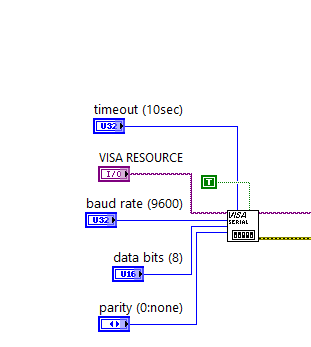


**Second part**

Serial communication Visa

1.

Initialisation of the visa serial:



To find the visa serial: search for Instruments I/O -> Serial-> configure port

2. visa open

search for instruments I/O->visa->advanced->open serial

open the session for exchanging variables in the serial.

3.visa write

search for instruments I/O->visa->advanced->write

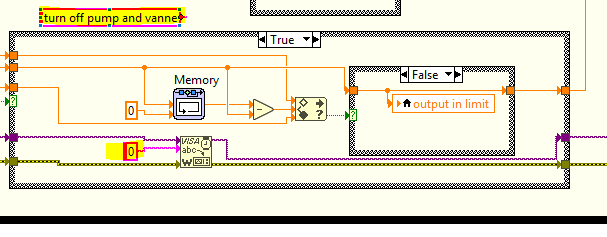
send a variable in the serial

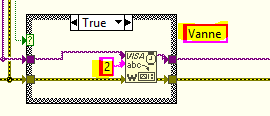
in our case we have three variables to sends

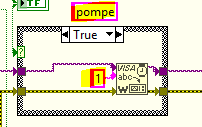
“1” -> indexing pump on

“2”-> indexing valve on

“0”-> both valve and pump are off







4.visa closed

search for instruments I/O->visa->advanced->close

close the session for exchanging and liberating the canal.

