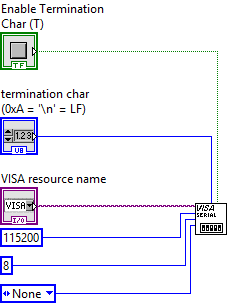
## Configuration



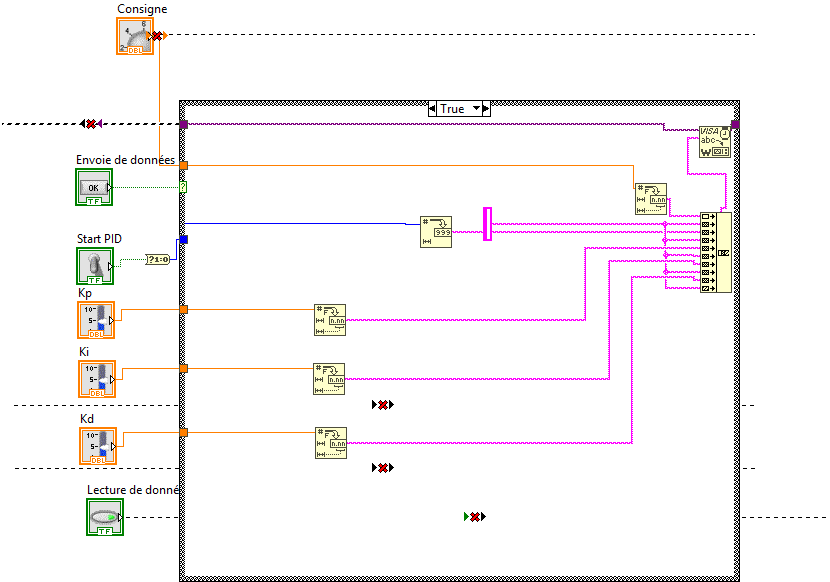
*Communication configuration blocks*

For the configuration part, we use the VISA configure Serial Port block, having downloaded the necessary drivers beforehand.

Then we create the control blocks and/or constants necessary for its proper functioning: in our case we will specify

* The Visa Resource Name (to choose the appropriate COM port)
* The activation of a termination character to separate the different messages during reception
* The baud rate which will be identical to the one used by the microcontroller.
* The existence or not of a parity bit
* As well as the size of the UART payload

## Writing from LabVIEW



*LabVIEW to STM32 write blocks*

In our case, we have several variables to transmit from LabVIEW to the microcontroller used:

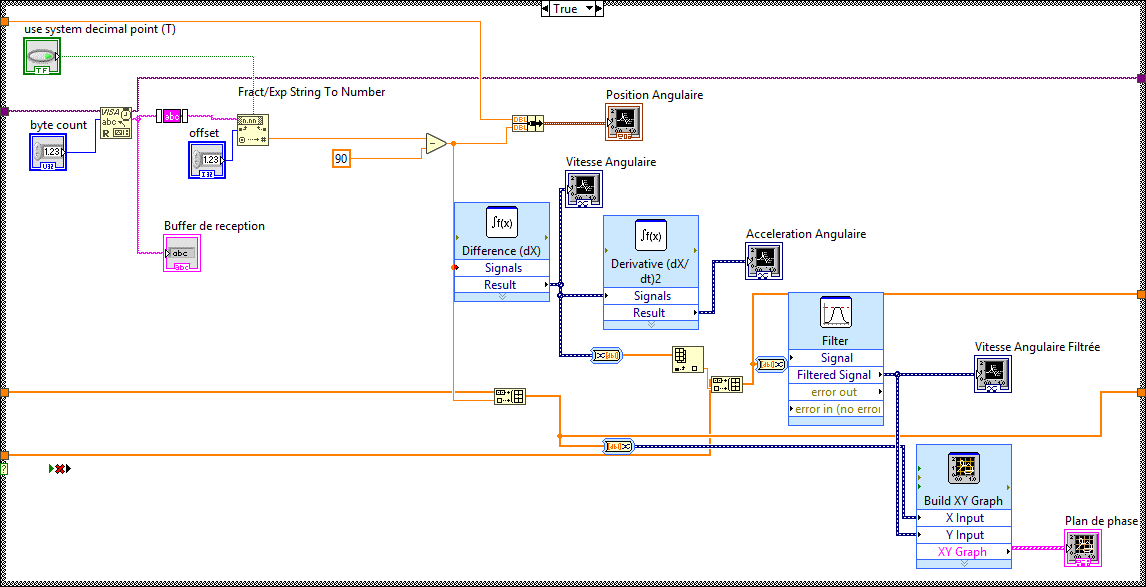
* A boolean variable indicating the start or stop of the system
* Real numbers containing the constants Kp, Ki and Kd of our implemented PID.
* A float containing the instruction for tests.

Therefore, it is necessary to separate the variables with " \n " to break up the message when receiving.

So our frame of data to be sent is a character string containing the variables to be sent separated by a predefined character " \n " where from the necessity to convert the variables into character strings and to carry out a concatenation before sending it.

It is necessary to implement on microcontroller an algorithm of decomposition of the sent frame to be able to extract the data.

## Reading from LabVIEW



*Blocks for reading data received from STM32*

LabVIEW is used in our case to receive the measured angular position on microcontroller.

To solve communication problems, we have deliberately sent the measured angle with an offset of + 90 degrees to ensure that the number is always positive.

This angle is processed in LabVIEW to be corrected and displayed on a graph.

From the received angles we drift twice to obtain respectively a speed and an angular acceleration.

We also perform a filtering on the calculated velocity to smooth the obtained curve.

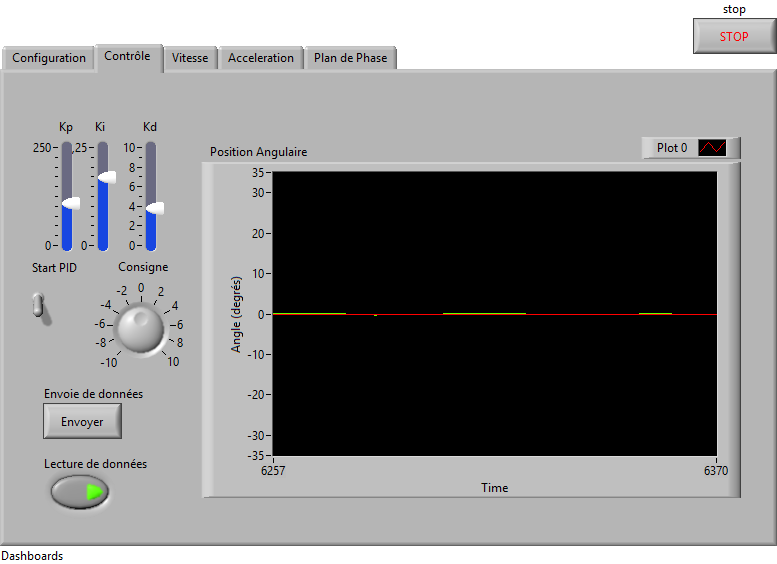
Moreover, we use the angular position and the angular velocity to plot the phase plane in real time by designating them respectively as X and Y.

## Final result

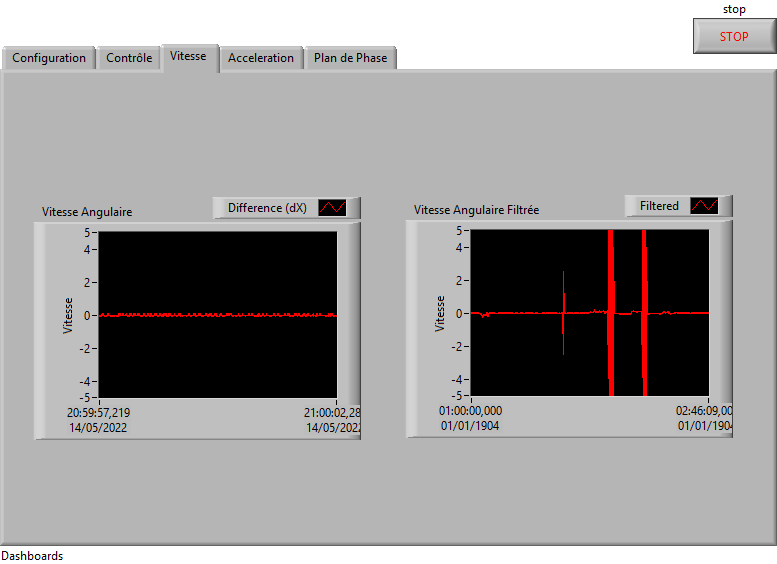
### **Interfaces**



*Screenshot of the control panel*

²

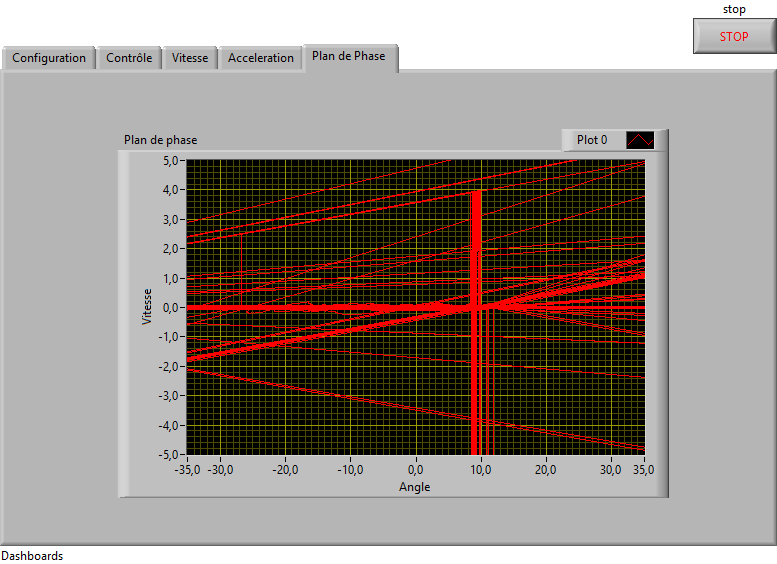
*PID control tab and angular position curve*



*Filtered and unfiltered speed panel*

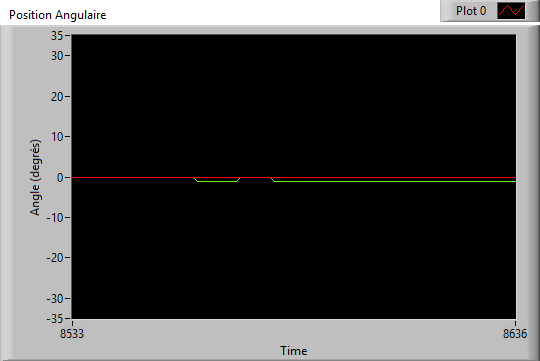


*Acceleration panel*

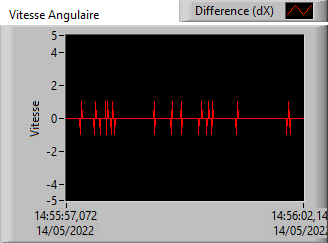


*Phase plan tab*

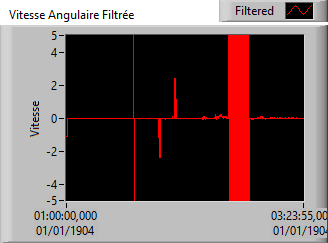
### **Real time plot**



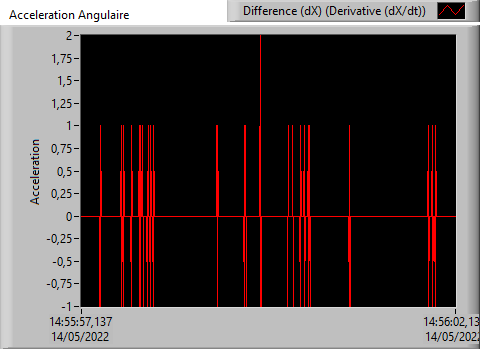
*Angular position time plot*



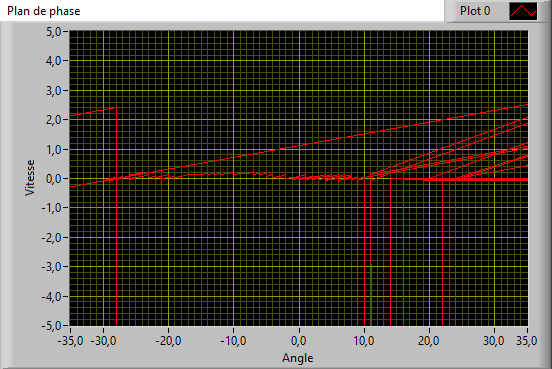
*Angular velocity curve*



*Curve of the filtered angular velocity*



*Angular acceleration*



*Phase Diagram*