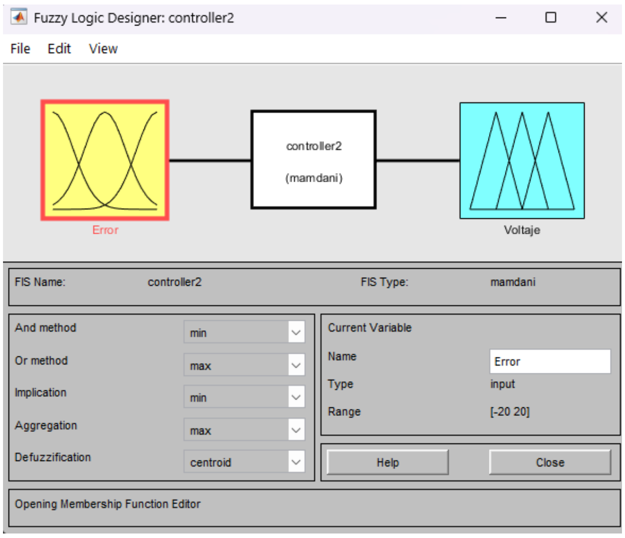
Fuzzy Controller

1. System implementation:

First, the Fuzzy controller was made with the “Fuzzy Toolbox” library with the parameters given in the slides, with 5 membership functions, two variables and 5 rules, as shown below:



FIS Name: controller2

FIS Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

Current Variable

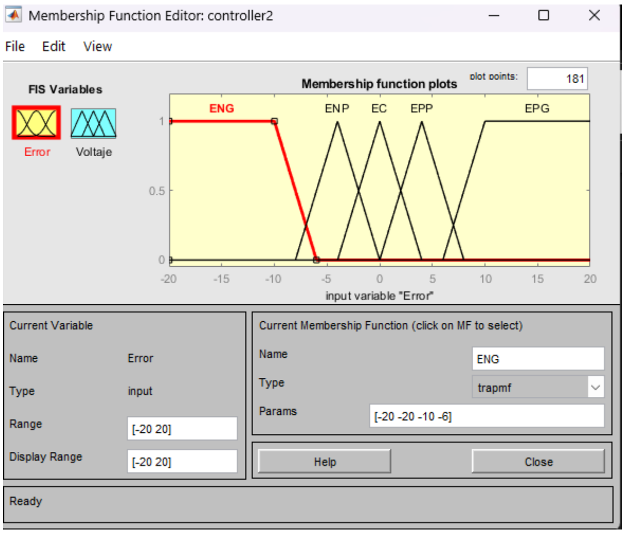
Name: Error

Type: input

Range: [-20 20]

Opening Membership Function Editor

Membership Features:



Current Variable

Name: Error

Type: input

Range: [-20 20]

Display Range: [-20 20]

Current Membership Function (click on MF to select)

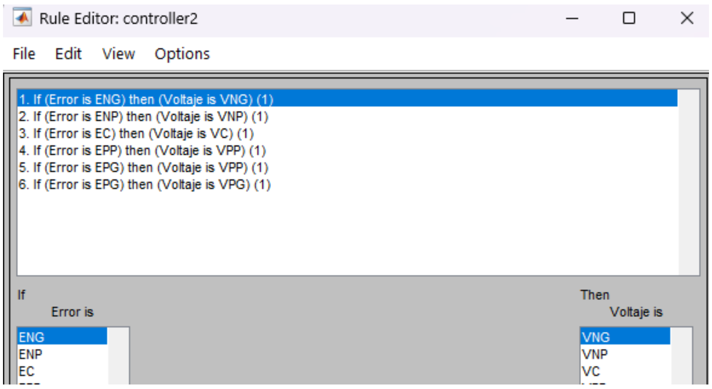
Name: ENG

Type: trapmf

Params: [-20 -20 -10 -6]

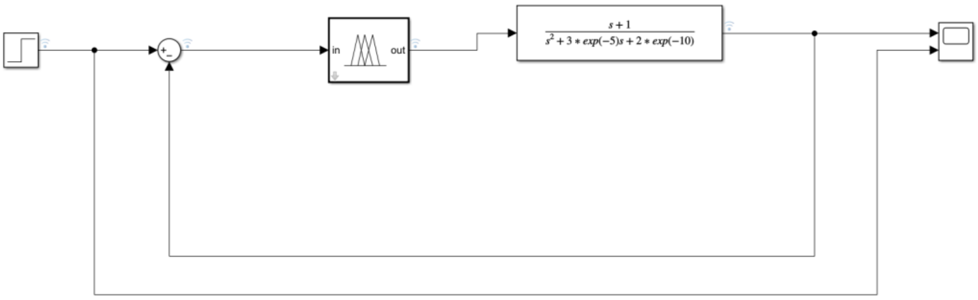
Ready

And the rules are:



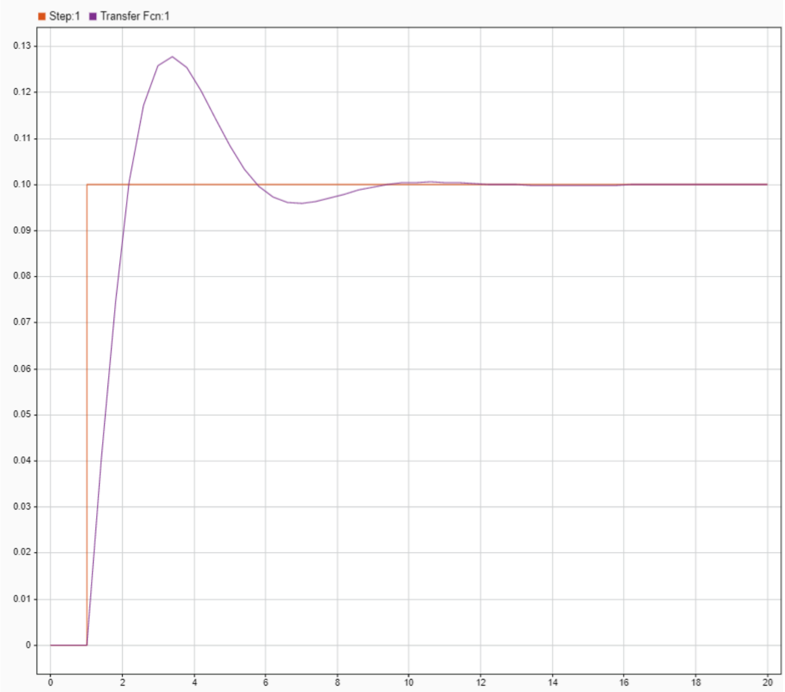
1. If (Error is ENG) then (Voltaje is VNG) (1)
2. If (Error is ENP) then (Voltaje is VNP) (1)
3. If (Error is EC) then (Voltaje is VC) (1)
4. If (Error is EPP) then (Voltaje is VPP) (1)
5. If (Error is EPG) then (Voltaje is VPP) (1)
6. If (Error is EPG) then (Voltaje is VPG) (1)

Now the implementation of the system in simulink is as follows:



1. Controller behavior:

As seen in the following figure, the behavior of the controller in the first 10 seconds is quite unstable and then it manages to establish itself in the desired position, it has quite a bit of overshoot and its stabilization is achieved approximately in 14 seconds.



The controller conditions are now defined in the following table:

Input Variables	1
Membership Features	5
Rules	5
Overshoot	High
Stabilization time	14 seconds

2. 3. Taking into account that the rules used in the implementation of fuzzy logic in the Engine are:

••If the error is large negative, then the voltage is large negative

•If the error is small negative, then the voltage is small negative

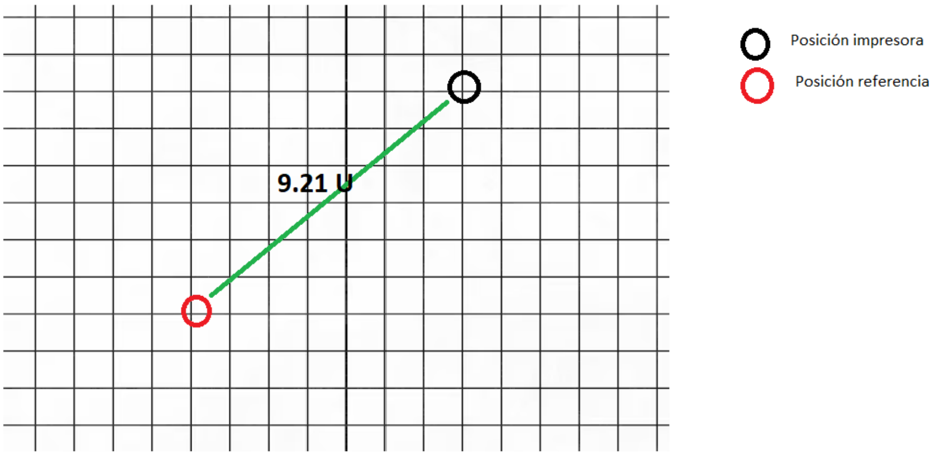
•If the error is zero, then the voltage is zero

•If the error is small positive, then the voltage is small positive

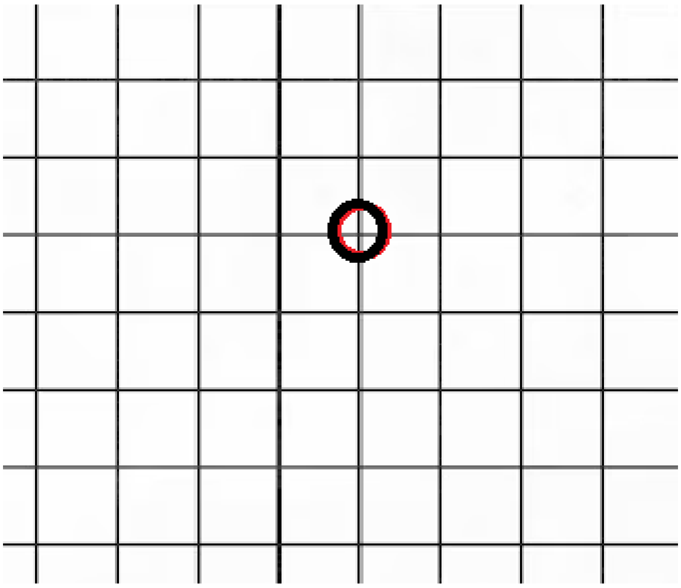
•If the error is large positive, then the voltage is large positive.

This logic is based on the differential distance between the printer location and the reference location, in the same way the motors act at a higher or lower speed depending on the voltage and its distance.

An example of this can be seen by having



In this case, the difference or error that appears would refer to the field of action of the ENG and ENP membership function, which would imply that at the output a VNG and VNP would be obtained in different percentages.



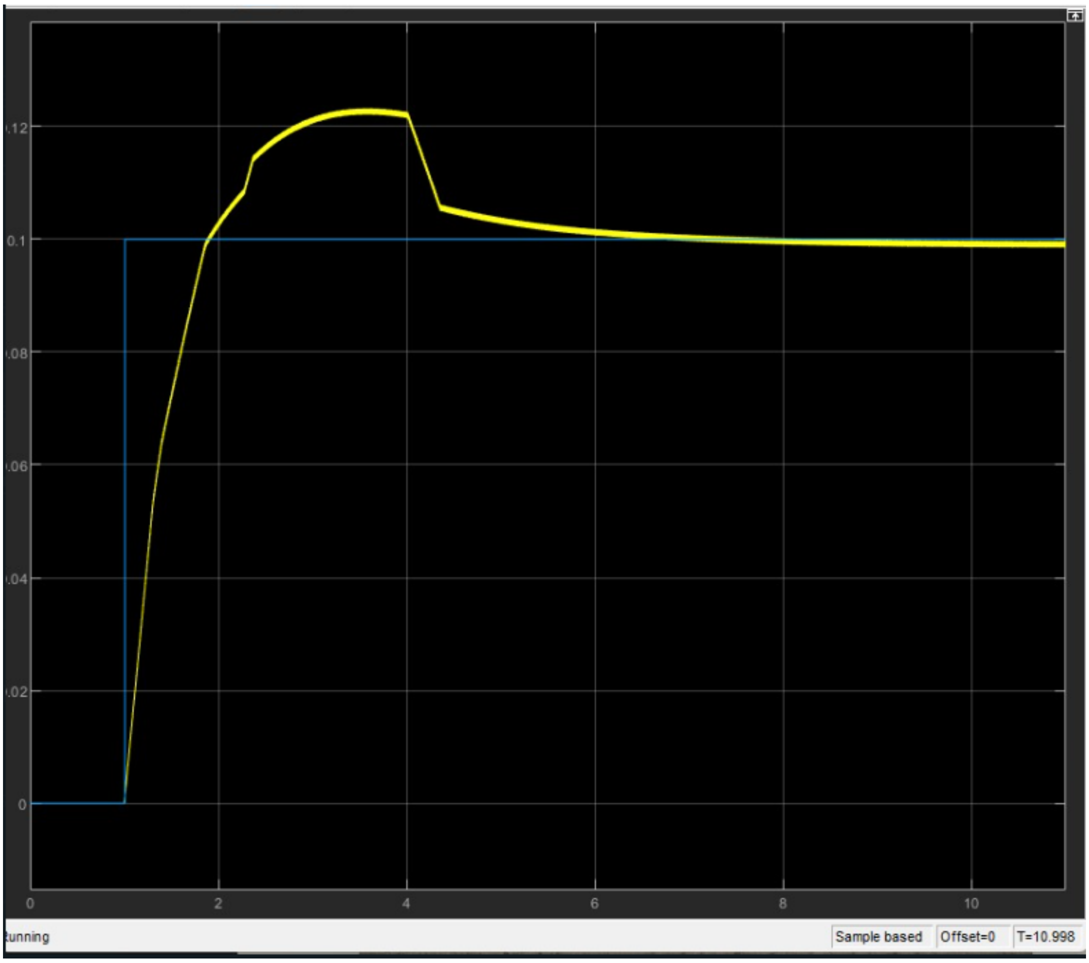
Another case that we can see is this one in which the reference position is the same or quite similar to the current one, which is why the error we would have is EC and the output with a VC. This means that there would be no movement unless some percentage was found in another membership function.

4. Adjust the fuzzy system to reduce controller overshoot and establishment.

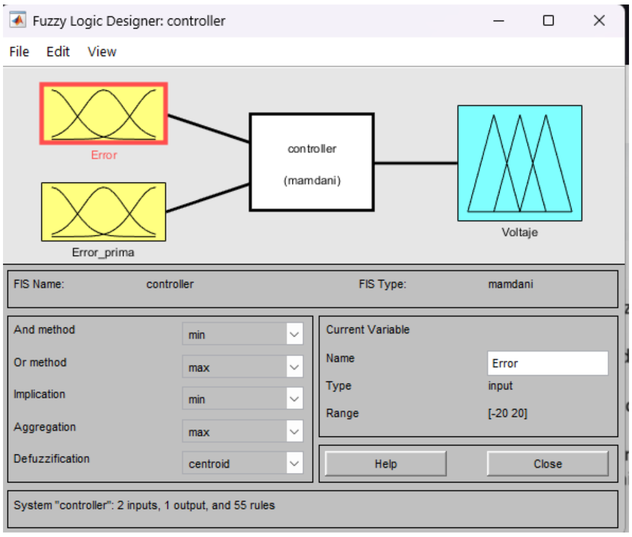
a. First test, it was carried out with the following conditions

Inputs	2
Membership Features	5
Rules	5
Overshoot	Alto
Stabilization time	8 seconds
Range of the Universe of the error_prima speech	[-5 up to 5]

The result is shown:



Implementation of the system, as seen, a variable derived from the error was added:



FIS Name: controller

FIS Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

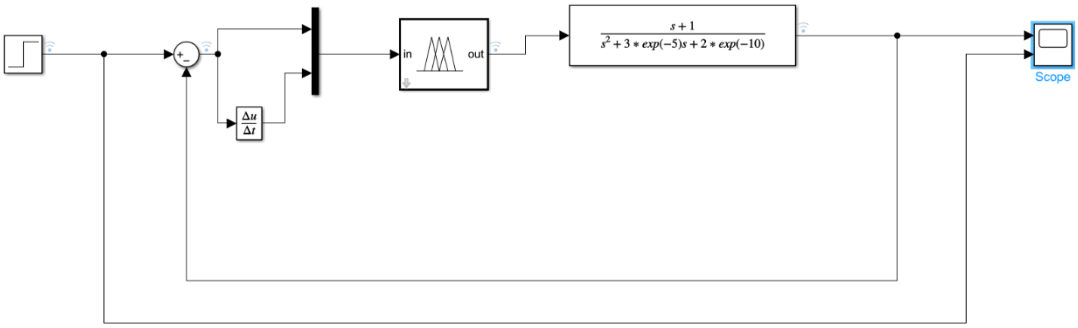
Current Variable

Name: Error

Type: input

Range: [-20 20]

System "controller": 2 inputs, 1 output, and 55 rules


+
-
$\frac{\Delta u}{\Delta t}$
in
out
$\frac{s+1}{s^2+3*exp(-5)s+2*exp(-10)}$
Scope

The conditions are the following::

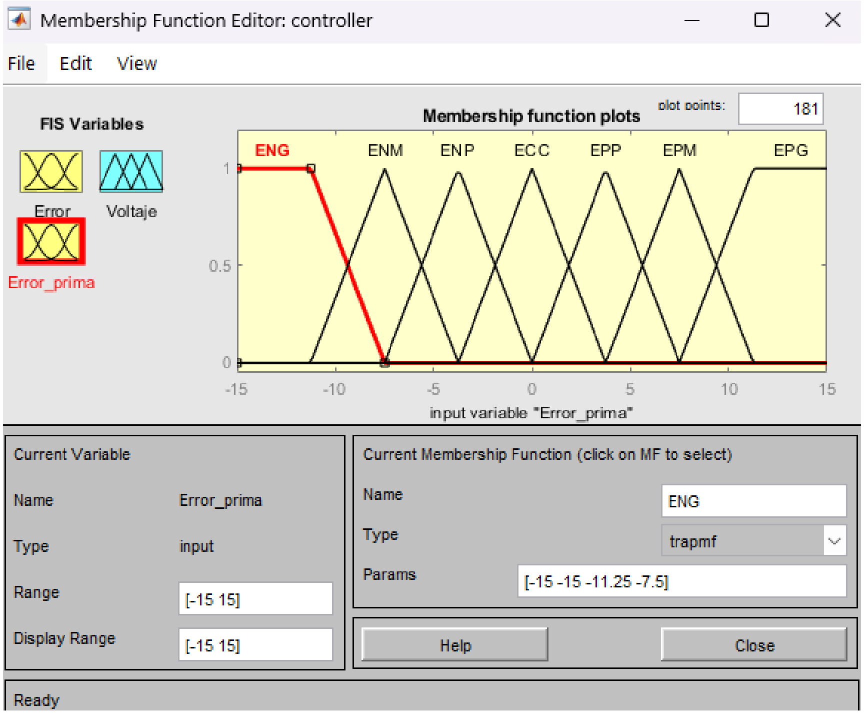
Inputs	2
Membership Features	7
Rules	54
Overshoot	bajo
Stabilization time	10 segundos
Range of the Universe of the error_prima speech	[-15 up to 15]

Now we tried more membership functions covering the universe of discourse for both inputs and output, as seen below. Reviewing the nomenclature of the functions first::

ENG	Error negativo grande
ENM	Error negativo medio
ENP	Error negativo pequeño
ECC	Error cero
EPP	Error positivo pequeño

EPM	Error positivo medio
EPG	Error positivo grande

Where we have ENM and EPM as the new membership functions compared to the other controllers



plot points: 181

Current Variable

Name Error_prima

Type input

Range [-15 15]

Display Range [-15 15]

Current Membership Function (click on MF to select)

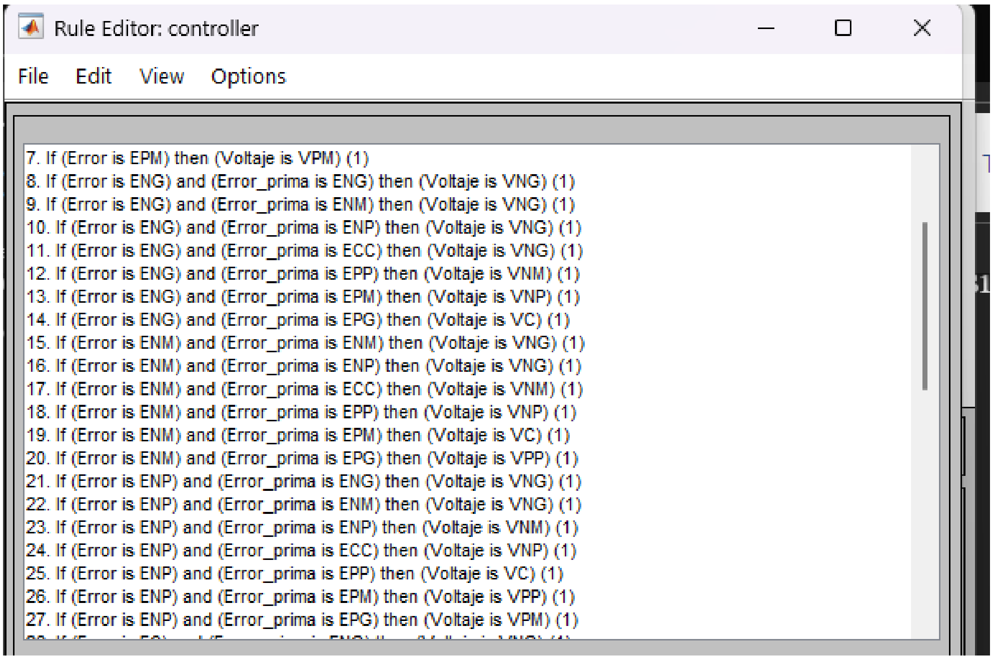
Name ENG

Type trapmf

Params [-15 -15 -11.25 -7.5]

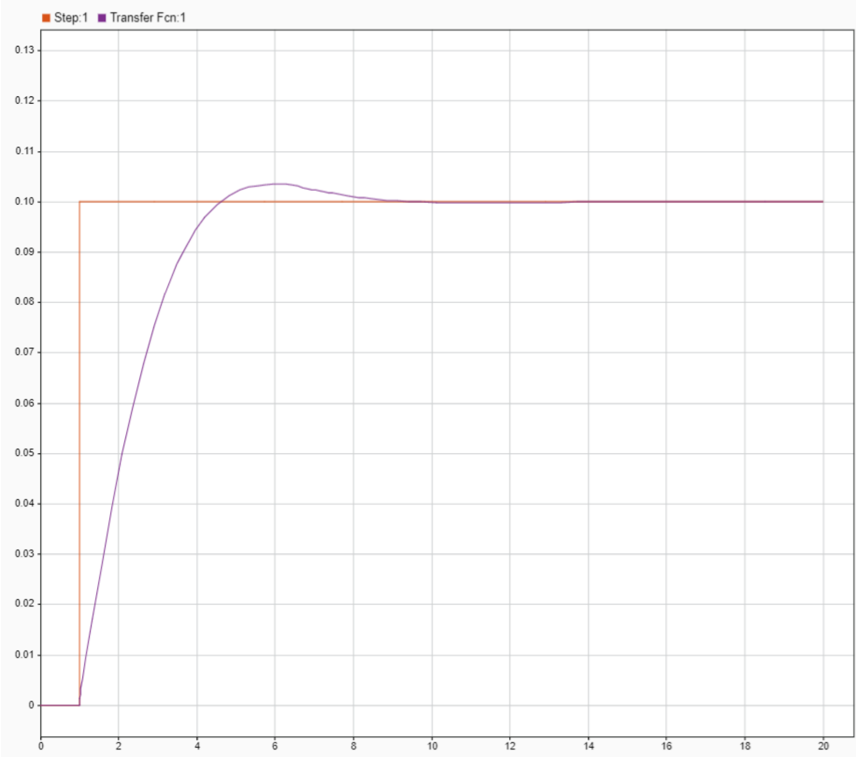
Ready

And a look at the added rules is shown



7. If (Error is EPM) then (Voltaje is VPM) (1)
8. If (Error is ENG) and (Error_prima is ENG) then (Voltaje is VNG) (1)
9. If (Error is ENG) and (Error_prima is ENM) then (Voltaje is VNG) (1)
10. If (Error is ENG) and (Error_prima is ENP) then (Voltaje is VNG) (1)
11. If (Error is ENG) and (Error_prima is ECC) then (Voltaje is VNG) (1)
12. If (Error is ENG) and (Error_prima is EPP) then (Voltaje is VNM) (1)
13. If (Error is ENG) and (Error_prima is EPM) then (Voltaje is VNP) (1)
14. If (Error is ENG) and (Error_prima is EPG) then (Voltaje is VC) (1)
15. If (Error is ENM) and (Error_prima is ENM) then (Voltaje is VNG) (1)
16. If (Error is ENM) and (Error_prima is ENP) then (Voltaje is VNG) (1)
17. If (Error is ENM) and (Error_prima is ECC) then (Voltaje is VNM) (1)
18. If (Error is ENM) and (Error_prima is EPP) then (Voltaje is VNP) (1)
19. If (Error is ENM) and (Error_prima is EPM) then (Voltaje is VC) (1)
20. If (Error is ENM) and (Error_prima is EPG) then (Voltaje is VPP) (1)
21. If (Error is ENP) and (Error_prima is ENG) then (Voltaje is VNG) (1)
22. If (Error is ENP) and (Error_prima is ENM) then (Voltaje is VNG) (1)
23. If (Error is ENP) and (Error_prima is ENP) then (Voltaje is VNM) (1)
24. If (Error is ENP) and (Error_prima is ECC) then (Voltaje is VNP) (1)
25. If (Error is ENP) and (Error_prima is EPP) then (Voltaje is VC) (1)
26. If (Error is ENP) and (Error_prima is EPM) then (Voltaje is VPP) (1)
27. If (Error is ENP) and (Error_prima is EPG) then (Voltaje is VPM) (1)

The controller output is now displayed:



Conclusion: We chose the last controller as the ideal one since it is the one with the best behavior, because it has very little overshoot and also its establishment time improves a lot. To carry out this controller, many more tests were carried out than those shown in this document, but the factors that we considered elementary to carry it out were adding: another input variable, more membership functions, and adding the rules according to a combinatorics (we use karnot). And also reduce the range of the universe of discourse of the variable derived from the error.