Control Science and Engineering
2023; 7(1): 1-11
http://www.sciencepublishinggroup.com/j/cse
doi: 10.11648/j.cse.20230701.11

Retrofit of a Governor Electronic-Analogic of a Francis Turbine of 180 MW of Decade of 1960 Using Microcontroller Arduino Due

Jose Luiz Guarino[1, *], Jose Flavio Silveira Feiteira[2]

[1]Furnas Centrais Eletricas, Rio de Janeiro, Brazil

[2]Computational Modelling in Science and Technology, Universidade Federal Fluminense, Volta Redonda, Brazil

Email address:

galloguarino@gmail.com (Jose Luiz Guarino), zeflavio@gmail.com (Jose Flavio Silveira Feiteira)

[*]Corresponding author

To cite this article:

Jose Luiz Guarino, Jose Flavio Silveira Feiteira. Retrofit of a Governor Electronic-Analogic of a Francis Turbine of 180 MW of Decade of 1960 Using Microcontroller Arduino Due. *Control Science and Engineering*. Vol. 7, No. 1, 2023, pp. 1-11. doi: 10.11648/j.cse.20230701.11

Received: January 9, 2023; **Accepted**: February 1, 20233; **Published**: February 16, 2023

Abstract: This paper describes the retrofit of a governor electronic-analogical of a Francis turbine of a synchronous generator of the decade of 1960. The author has many years of experience in maintenance in governors of hydroelectric power plants including excitation systems. Governors are responsible to maintain constant the frequency of a power electric net. The original model of the governor in question was obtained in local field tests, not related in this paper. In this retrofit, time constants were maintained to respect the original project and avoid hydraulic harmful and dangerous transitory problems. The original continuous model of the governor was simplified toa a second order model. The second order continuous model was emulated to a discrete system. The difference equation was created and microcontroller Arduino Due was programming. The digital controller produced, was syntonized to cancel the mathematical function of the servomotor, to reduce the grade of the resulted system obtaining a second order system. The hydraulic part of the governor was simulated using 741 operational amplifiers. During the tests in the workbench the author burned two Arduino's electronic cards so it was necessary to create the electronic interface between Arduino and the 741 amp op including over and under voltage protection. The complete set was successfully tested in workbench. The mathematical models were priory simulated with Matlab-Simulink and there were the corresponding electronic simulations in workbench that confirmed the results. This retrofit solution, using Arduino Due, is of low cost compared to traditional manufacturers.

Keywords: Governor, Retrofit, Arduino Due

1. Introduction

It is known that a governor of a hydraulic turbine of a synchronous generator has the function to maintain the velocity of the rotation constant to difference electric loads of the generator, responsible to maintain the electric frequency net constant, figure 1. The governor electronic-analogical of decade of 1960 was originally built with discrete components what demanded much physical space, like a refrigerator. We propose a retrofit using microcontroller Arduino Due, preserving the original constants of time of the governor. This microcontroller is employed in stereo audio applications because he has two analogical inputs and two analogical outputs.

We simplified the original continuous model to a second order model an then we emulate to a discrete model using the z-Transform, then we generated the difference equation. Finally, we programed the Arduino Due. We simulate the hydraulic part of governor using 741 operational amplifiers. Interfaces between Arduino Due and the 741s were needed including over and undervoltage circuit protection because of difference level of voltages. We executed step entries both in the mathematical models, using Matlab-Simulink, and in the workbench set and we realized that we obtained success in the retrofit. The final physical size of the retrofit is just of a shoe-box. The author has worked in maintenance of governors for many years and has observed two kinds of professionals, one just theorical, and another just practical. The major purpose of this work is the dialogue between practice and theory. [1-3, 7, 15].

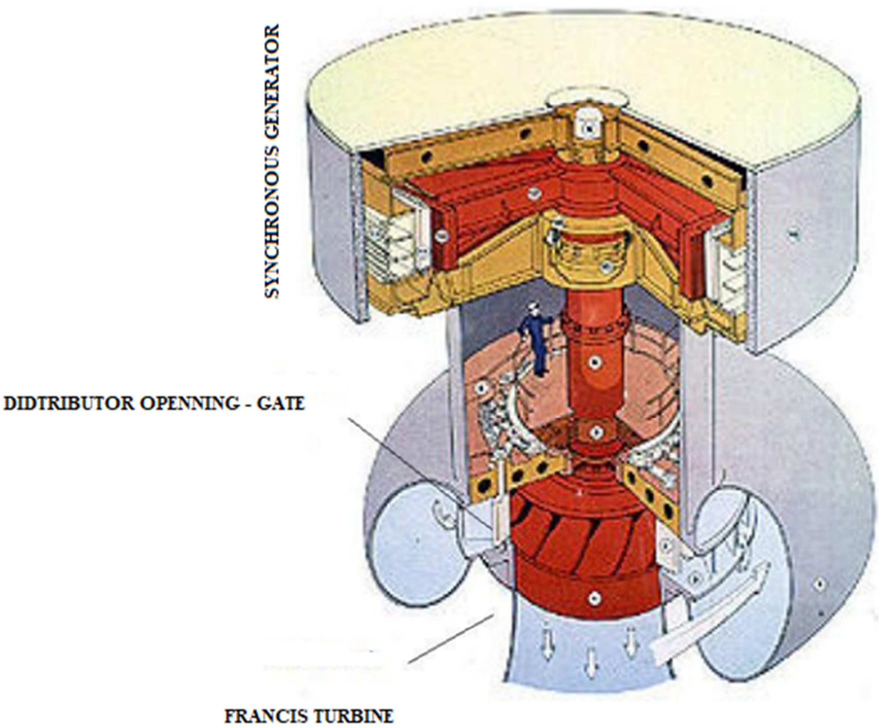

Figure 1. *Turbine Francis and generator set.*

2. Original Mathematical Continuous Model of the Governor

Figure 2 shows the mathematical model, in Laplace terms, of the original electronic-hydraulic governor that will suffer the retrofit. [1, 4, 15].

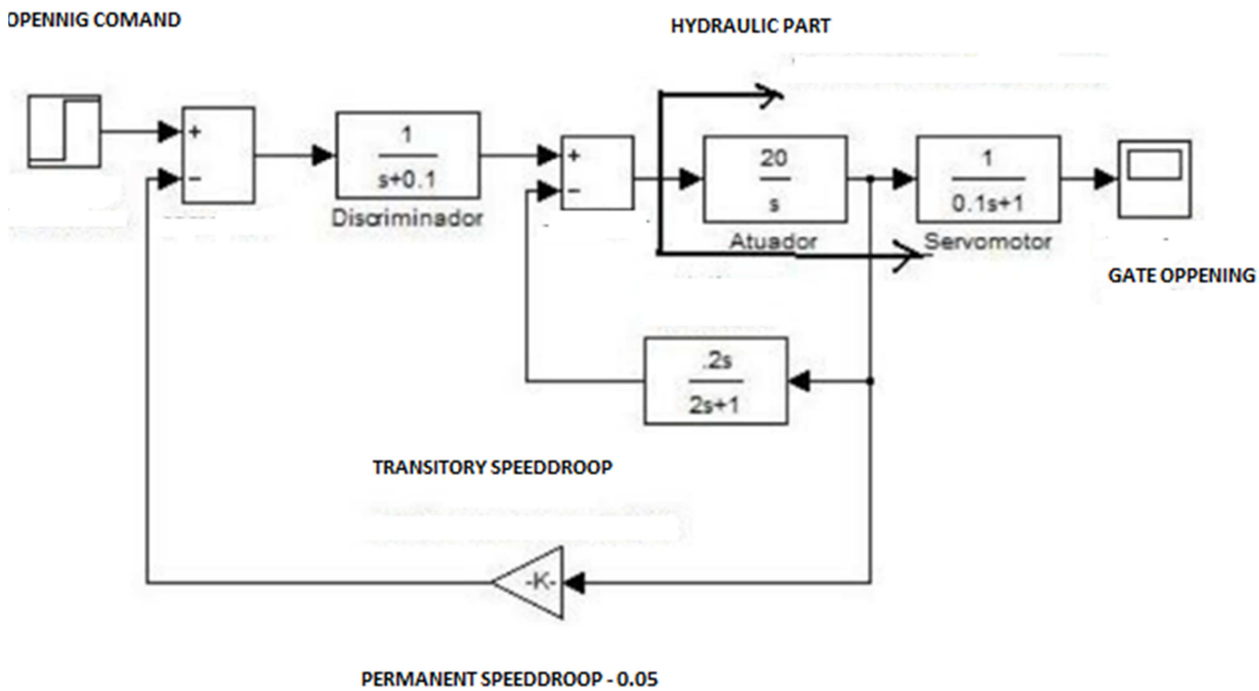

Figure 2. *Mathematical model of the governor.*

Figure 3 shows the simulation of step response of 100% of the mathematical model of figure 2, using Simulink. [1, 2].

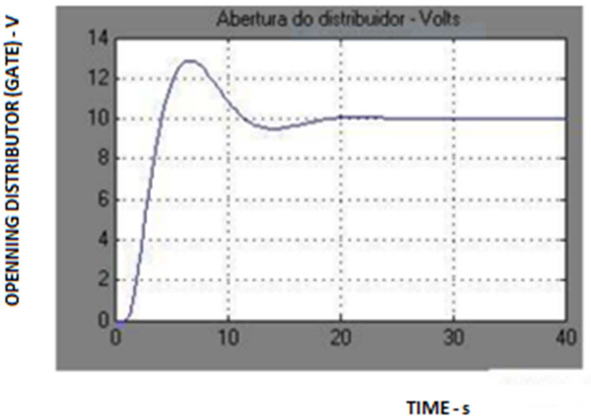

Figure 3. Step response of the governor of 100%.

3. Second Order System

There is a rule of thumb that says that every system of higher order can be reduced to a second order system. Figure 4 shows a second order system and Figure 5 shows the step response of 100% of this system. [1, 4, 6].

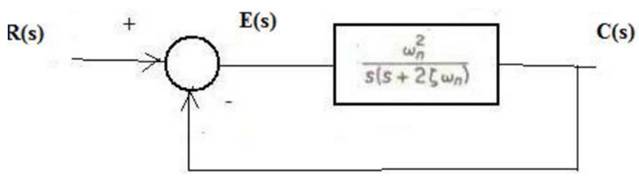

Figure 4. Second order system.

Comparing figure 3 and figure 5 we estimate:

$$.\zeta = 0.35 \quad (1)$$

Looking to figure 3 we estimate the time of stabilization, t_s is:

$$ts = \frac{4.6}{\zeta.\omega n} = 25\ s \quad (2)$$

Using (1) and (2) we find:

$$.\omega_n = 0.52 \text{ rad/s} \quad (3)$$

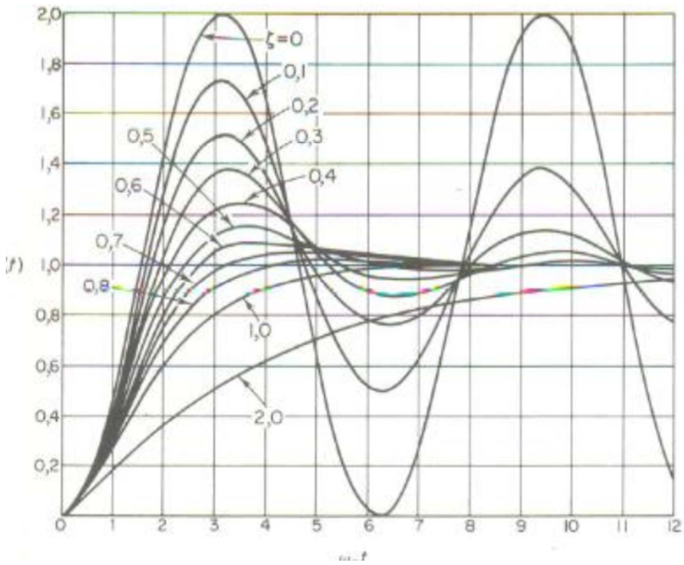

Figure 5. Steps response (100%) second order system.

4. New Equivalent Syntonized Model of the Original Model of the Governor

Figure 6 shows the new syntonized transfer function G to cancel the transfer function of the servomotor. The step 100% response was preserved both n time and in shape, as calculated in paragraph 3. [1, 4].

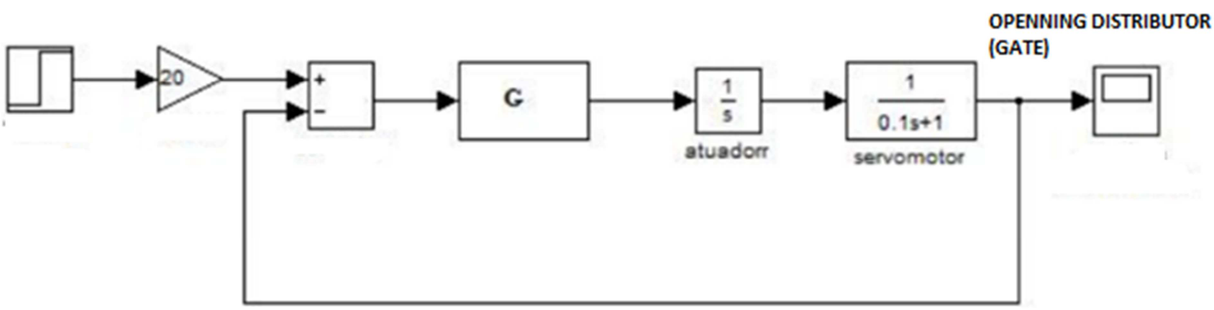

Figure 6. New model of the governor.

Based on figure 4 and (1) and (3), we make G syntonized to cancel the servomotor transference function, so G becomes:

$$G = 0.52^2.(0.1s + 1)/(s + 0.36) \tag{4}$$

The new second order equivalent model of the governor is shown in figure 7.

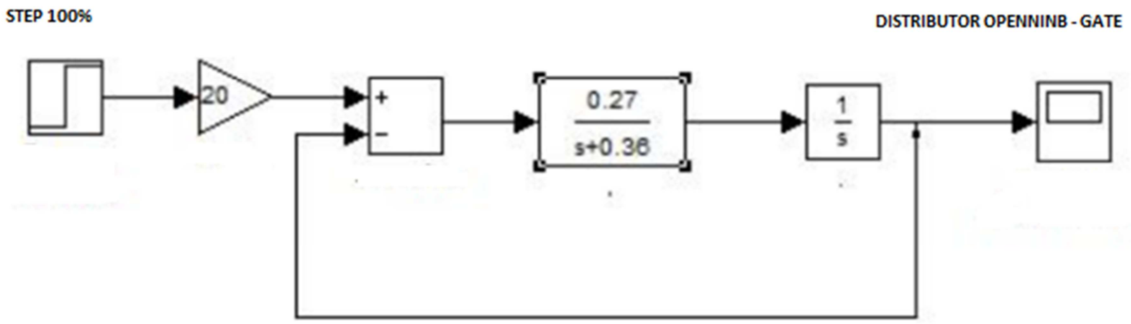

Figure 7. *New second order equivalent model of the governor.*

5. Discrete Time Equivalent Model of the Reduced Second Order New Model of the Original Governor

We will discretize the function G(s) to the function D(z). This technique is known as emulation. The function D(z) is given for:

$$D(z) = k(z - z1)/(z - z2) \tag{5}$$

$$z = e^{sT} \tag{6}$$

We choose:

$$T < 25/100\, T = 0.2\, s \tag{7}$$

Looking to G(s), given by (4) the zero is s_1 = -10, so using (6) and (7) we have:

$$z1 = e^{-10.0.2} = 0.1353 \tag{8}$$

In the same way, the pole of G(s), s_2 = -0.36, so:

$$z2 = e^{-0.36.0.2} = 0.9305 \tag{9}$$

So, using (8) and (9), (5) becomes:

$$D(z) = k.(z - 0.1353)/(z - 0.9305) \tag{10}$$

But for s = 0, from (6), z = 1, and (4) and (10) are equivalent, then:

$$k.(1 - 0.1353)/(1 - 0.9305) = 0.27.(0.1.0 + 1)/(0 + 0.36) \tag{11}$$

$$k = 0.06 \tag{12}$$

Then D(z), the digital controller, becomes:

$$D(z) = (0.06.z - 0.0082)/(z - 0.9305) \tag{13}$$

Then, our new digital model becomes:

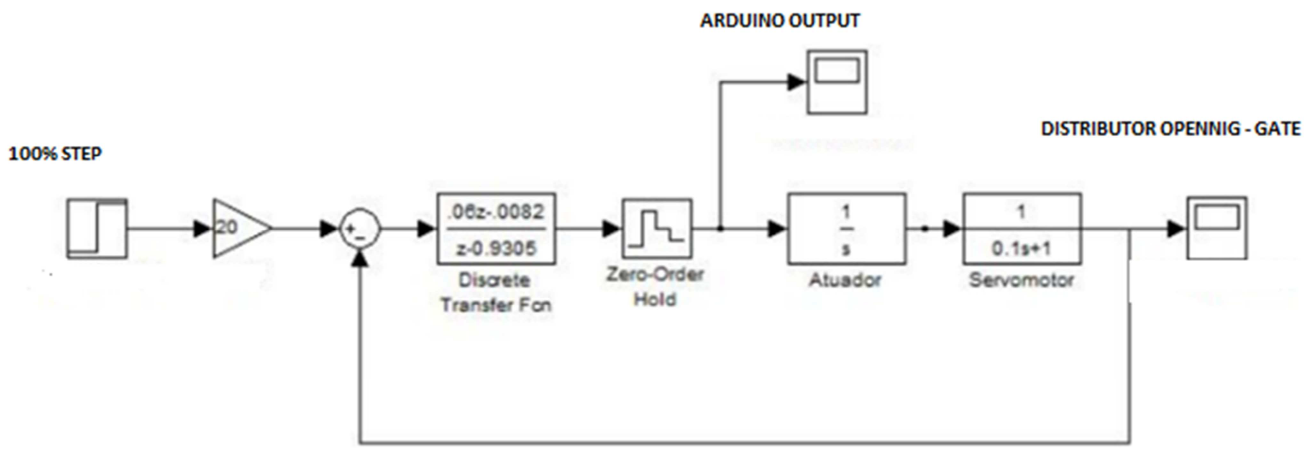

Figure 8. *Digitalized new model of the original governor.*

The step 100% response of the governor and the digital controller are: figures 9 and 10.

Comparing figure 9, the digitalized governor to figure 3, the original one we find them similar. [1, 5, 8].

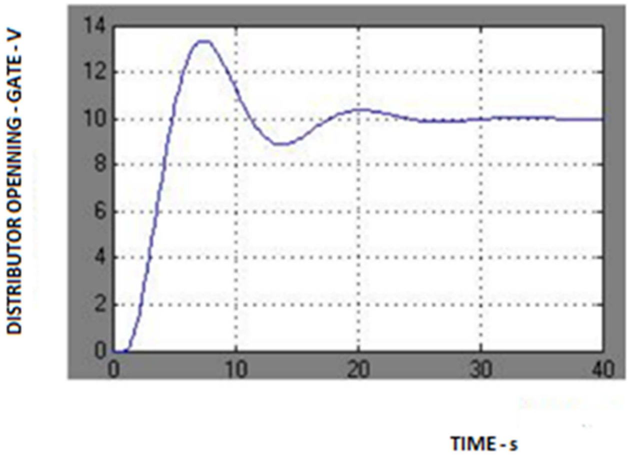

Figure 9. *Output of the digital governor.*

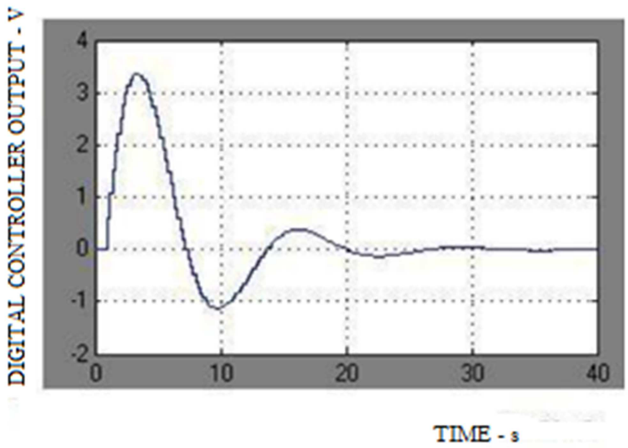

Figure 10. *Output of the digital controller of the new governor model.*

6. Difference Equation and Programing the Arduino Due

Figure 11 shows a schema of the digitalized governor.

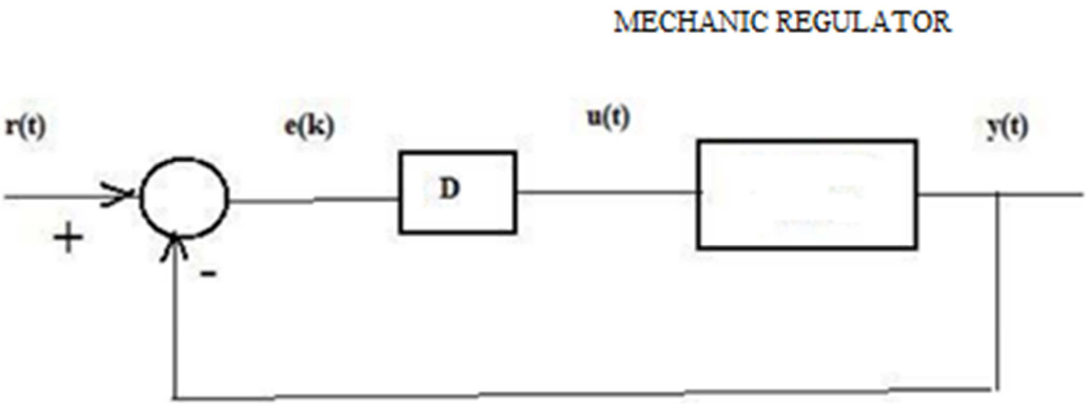

Figure 11. *Digitalized governor.*

From (13) and figure 11 we have:

$$D(z) = u(z)/e(z) = (0.06 - 0.0082z^{-1})/(1 - 0.9305z^{-1}) \tag{14}$$

Or:

$$u(z).(1 - 0.9305z^{-1}) = e(z)(0.06 - 0.0082z^{-1}) \tag{15}$$

Calculating the Inverse z-Transform of (15), we find the following difference equation:

$$u(k) = 0.9305.u(k-1) + 0.06e(k) - 0.0082e(k-1) \tag{16}$$

Looking to figure 11, for the programming of the Arduino Due we use the auxiliar equations:

$$.u = u' + 0.06.e \tag{17}$$

$$.u' = 0.9305.u_{old} - 0.0082.e_{old} \tag{18}$$

$$.e = r - y \tag{19}$$

The program code of the Arduino Due is shown in figure 12.

Algoritmo 4 Codigo Computacional do Controlador Digital Implementado com o Arduino Due $T = 0,2$ segundo

```
    // Codigo Computacional do Controlador Digital do Regulador de Velocidade T=0.2
    segundo
    double uold = 0; //Inicialização das variáveis
    double eold = 0;
4:  double e = 0;
    double r = 0;
    double ri = 0;
    double y = 0;
8:  double yi = 0;
    double u = 0;
    double uli = 0;
    int uaux = 0;
12: const int analogInPin0 = A0; //Pino da referência ri
    const int analogInPin1 = A1; //Pino de yi
    void setup ( )
    {
16: pinMode (analogInPin0,INPUT); //Portas de Entrada e Saída
    pinMode (analogInPin1,INPUT);
    pinMode (DACO,OUTPUT);
    analogReadResolution(12);
20: analogWriteResolution(12);
    }
    void loop ( ) { // Começa o laço de controle
    ri = analogRead(analogInPin0);
24: r = ri*10/4095;
    yi = analogRead(anaqlogInPin1);
    y = yi*10/4095;
    e = r - y;
28: u = uli + 0.06*e;
    uaux = int((2047/3.4)*u + 2047));
    analogWrite(DAC0,uaux); //Saída do controlador digital
    uold = u;
32: eold = e;
    uli = 0.9305*uold - 0.0082*eold;
    delay(200); //Espere 0,2 s
    }
```

Figure 12. *Programing code of the Arduino Due.*

As can be seen in line 34 of the code, we use T to 0.2s. In line 26 the number 10, represents the voltage of the full opening (100%) of the distributor, the output of the governor. The number 4095 in line 24 in the code is because of the number of steps of the conversion A/D of the Arduino Due. The number 3.4 in line 29 is the total output in voltages of the conversion D/A. [1, 5, 8-11].

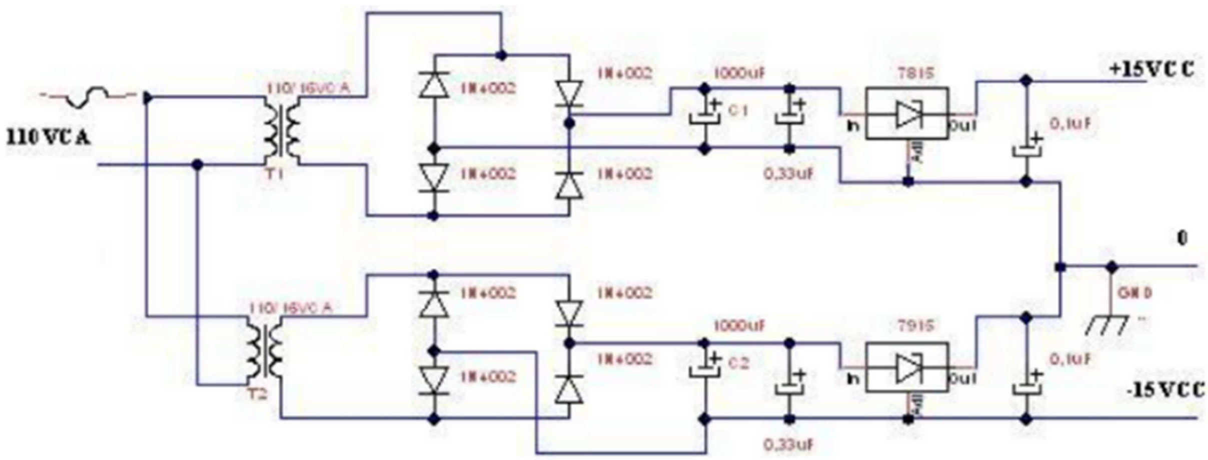

Figure 13. *DC Source of +/-15.*

7. Electronic Interface of the Arduino Due, Sources and f the Simulator Hydraulic Part of the Governor

Initially we show the circuit of the source of +/- 15 VDC, figure 13, to feed the interfaces and the electronic simulator of the hydraulic part of the governor that use the 741 amp ops. The ICs 7815 and 7915 should use radiators.

Then transformers are of 110/16 VAC, 25 VA.

Arduino Due should be feed with DC voltage between 7 and 12, we choosed 8 VDC. Figure 14 shows the circuit to feed the Arduino Due. The IC 7808 should use radiator.

Figure 15 shows the original model of the governor, compare to figure 2, using amplifiers, integrators and adders.

We will use model of figure 15 except the gain of the actuator of 20, see figure 8, to synthesize the hydraulic part of the governor using 741 amp ops, as can be seen in figure 16.

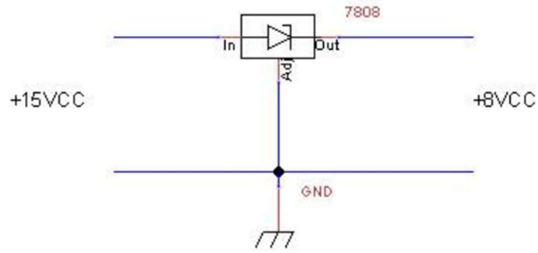

Figure 14. *The DC source of Arduino Due.*

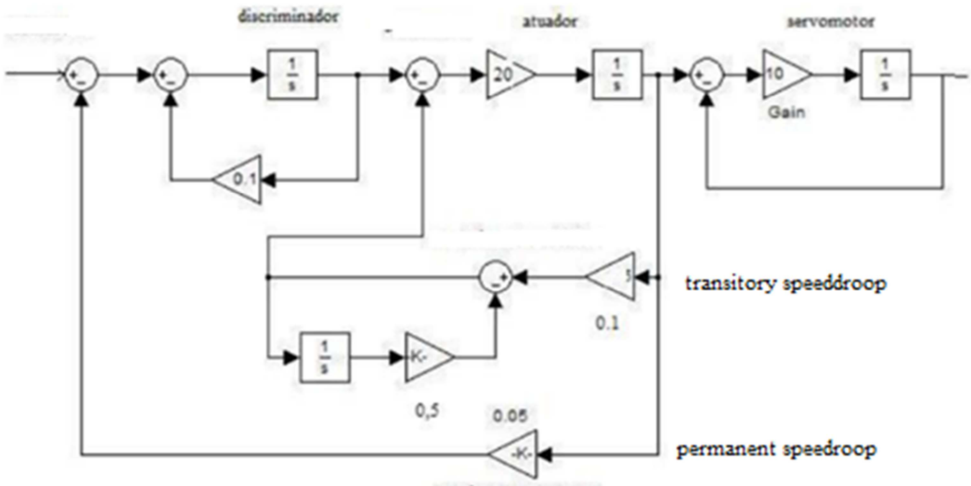

Figure 15. *original model.*

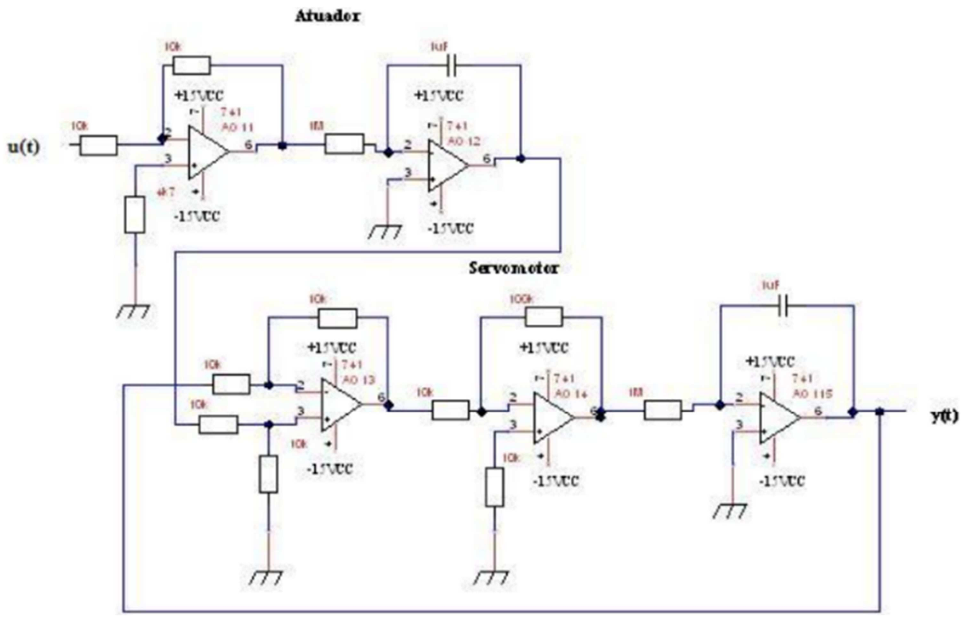

Figure 16. *Hydraulic part of the governor.*

NOTE: ALL ELECTRONIC CIRCUITS IN THIS PAPER, USING THE 741 AMP. OP. EMPLOYE NULL POTENTIOMETERS, NOT SHOWN, IN THE DIAGRAMS THAT COULD OVERCHARGE THEM.

Figure 17 shows the interface and protective overvoltage and undervoltage circuit of the first analogical input port (A0) of the Arduino Due. The potentiometer is set to 100 mV, to give a step input of 20%. The fusible is of 100 mA and the diode 1N746 is a Zener of 3.3 V.

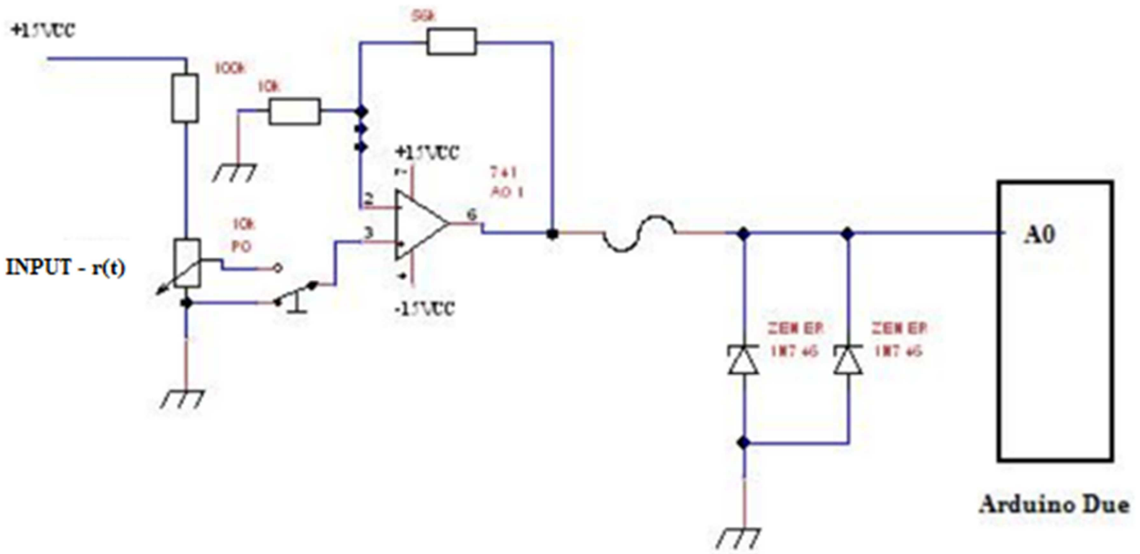

Figure 17. *Interface and protective input of Arduino Due.*

Figure 18 shows the interface and the protective circuit (similar to A0) of the feedback input of the second analogical port input of Arduino Due (A1). That is the reason why this Arduino is known as Due (two in Italian language).

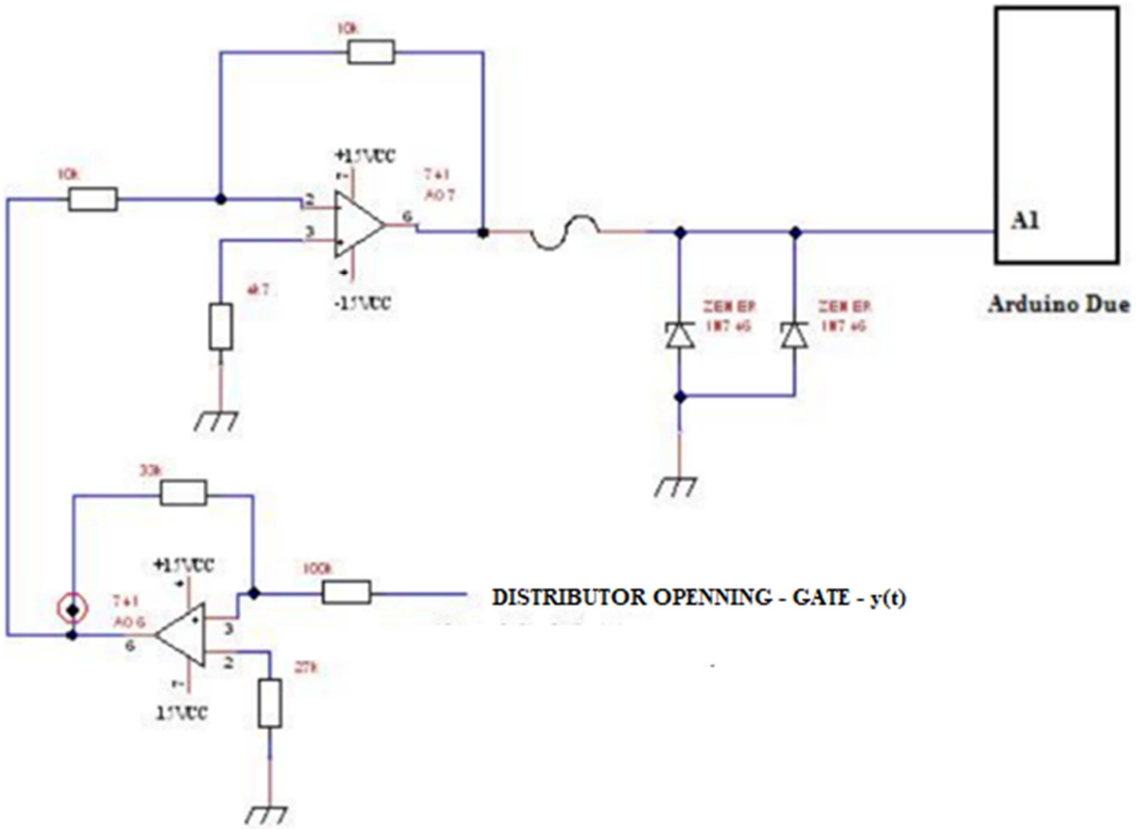

Figure 18. *Interface and protective feedback input of Arduino Due.*

Figure 19 shows the interface output port (DAC0) of Arduino Due. This port converts the digital output to analogical output between 0,55VDC and 2,75VDC.. The signal y(t), opening of the distributor, must varies between 0 and 10 VDC. We remember that the feeding of the 741 amp op is +/- 15 VDC so the necessity of interface.

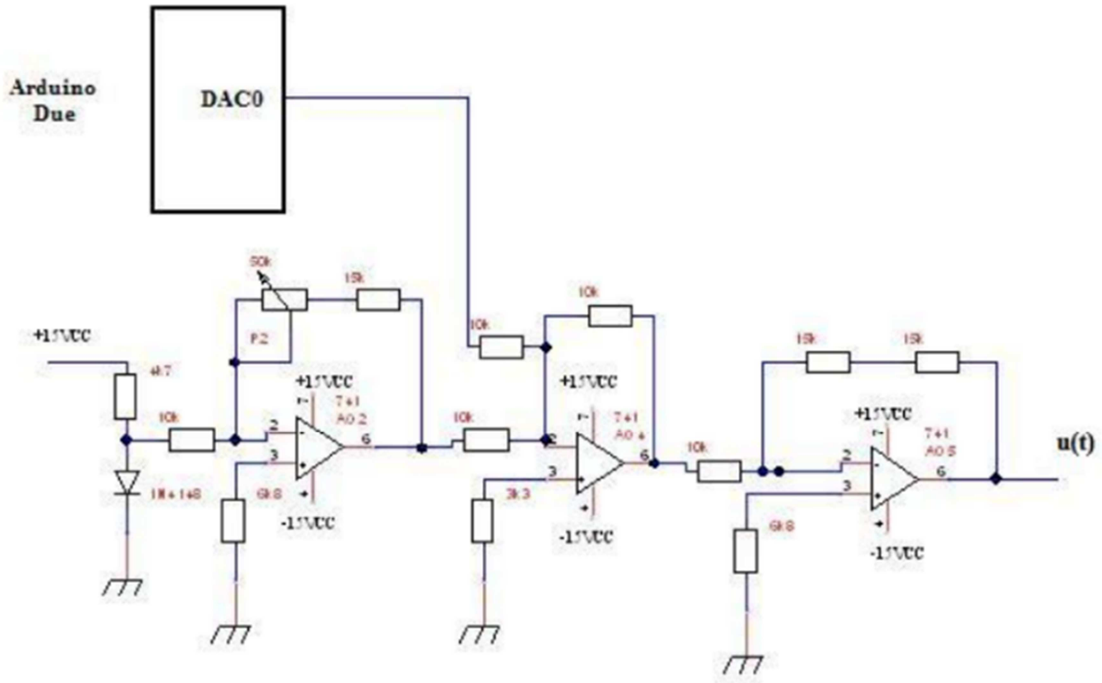

Figure 19. *Interface output of Arduino Due.*

Figure 20 shows the complete circuit set (inputs and output interfaces and protective circuits) of Arduino Due. [1, 6, 10, 11, 12, 13, 14].

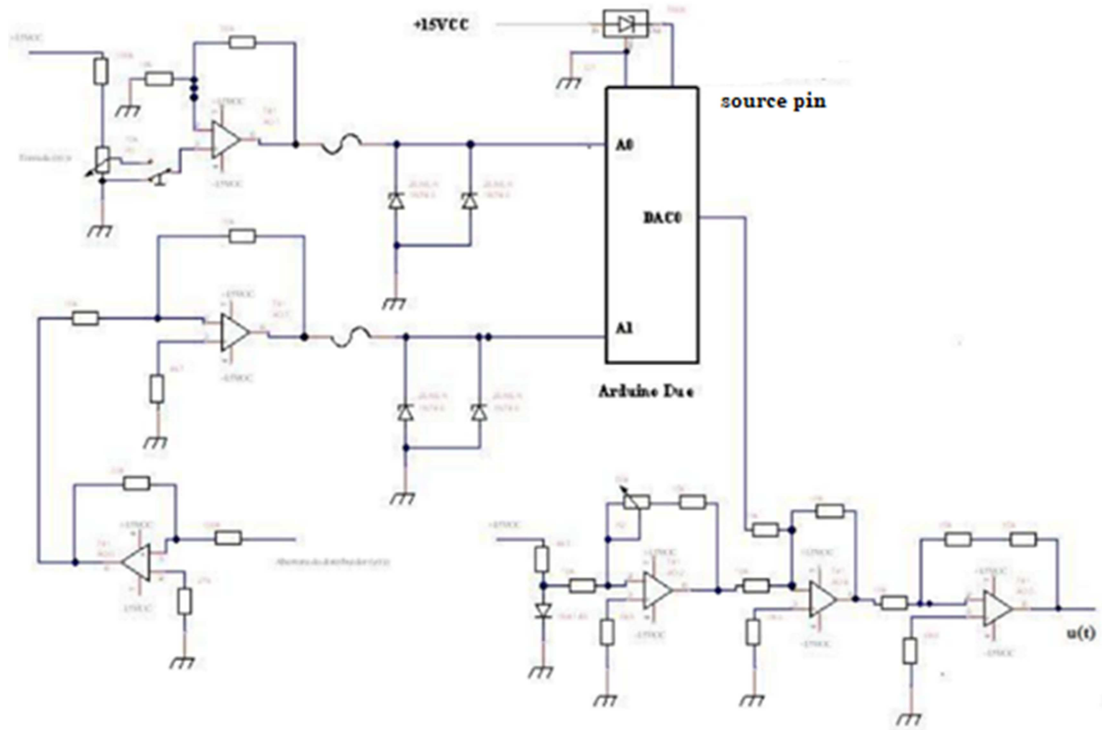

Figure 20. *Complete inputs and output set of Arduino Due.*

8. Tests and Results

Figure 21 shows the simulation of the response of the signal u(t) to a step input of 20% of the model of figure 8.

Figure 22 shows the sane simulation and the response signal y(t), opening of distributor.

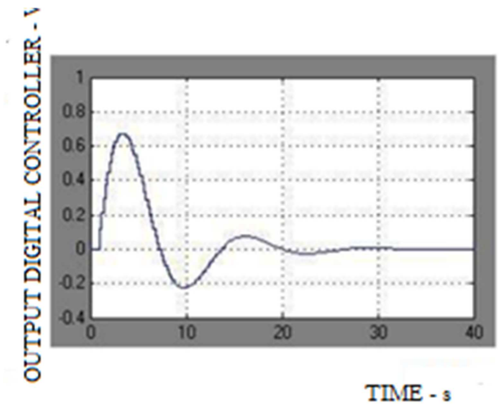

Figure 21. *Signal u(t) to a step input of 20%.*

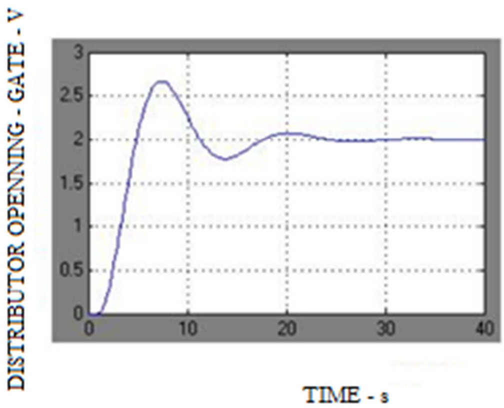

Figure 22. *Signal y(t), distributor opening to a step input of 20%.*

Figure 23 shows the photo of the complete retrofit of the governor in the workbench, Arduino Due, the smallest card (the green one), the hydraulic simulator, the smaller card and the interface, the big one.

Figure 23. *The retrofit of the governor.*

Figure 24 shows the linearity of input (reference, r)/output (distributor opening, y) of the entire circuit of the retrofit of the governor.

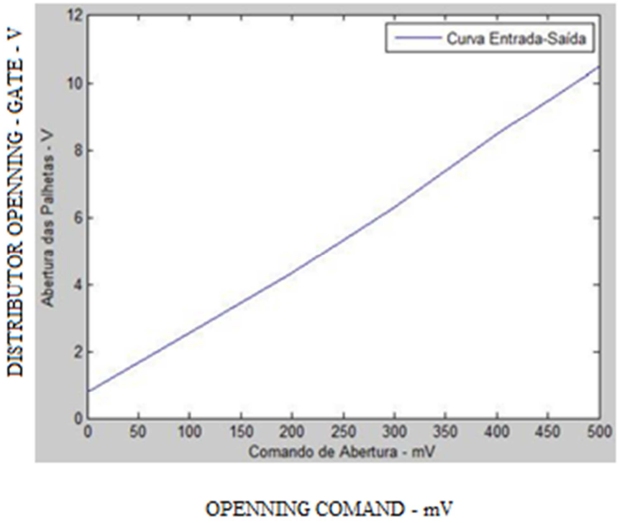

Figure 24. *Linearity of input-r/output-y of the retrofit.*

Figure 25 shows the photo of the screen of the oscilloscope after the step input of 20% of the retrofit of the governor. [1].

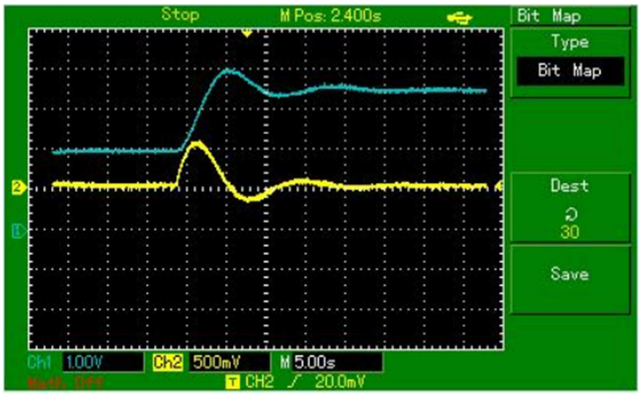

Figure 25. *The step response of 20% of the retrofit of the governor.*

9. Conclusion

Comparing figures 21, 22 and 25 we find that we achieved success in the retrofit. Additional program routine is needed to start with assurance the Arduino Due to execute the control. The reference, r can be also internally digitally adjusted. The program code shown in figure 12, was registered in the Brazilian Bureau of Patents – INPI (Instituto Nacional de Propriedade Industrial). This retrofit solution is of very low cost compared with traditional manufacturers and the original physical size was reduced from a refrigerator to a shoe-box [1, 5, 8].

Acknowledgements

I thank first to the Divine Providence, that helped me throughout my career of student and electric engineer at

Furnas Centrais Eletricas to conquer my know-how. I thank too to my mother, Margarida, my wife Marcia and my daughter, Mariana, that helped and aided me with their support and love. I cannot forget specially my colleagues that work with me with maintenance of governors and excitation systems of synchronous generators, from whom I learned a lot and without their experience and skills the writing of this paper would be impossible.

References

[1] Jose Luiz Guarino. Modelagem e Digitalização de Regulador de Velocidade Eletrônico-Analógico da Década de 1960 de Turbina Hidráulica Francis de Gerador Síncrono. 2014 http://mcct.uff.br select: Documentos/Teses, Dissertações, number 15.

[2] F. P. de Mello. Dinâmica e Controle da Geração, Curso de Engenharia em Sistemas Elétricos de Potência, serie PTI Convenio Eletrobrás/UFSM, 1979.

[3] Fitzgerald & Kingsley. Electric Machinery. Mc Graw Hill, 1980.

[4] Ogata. Modern Control Engineering. Prentice-Hall, Inc., 1982.

[5] Gene F. Franklin, J. David Powell, Michael L. Workman. Digital Control of Dynamic Systems. Addison-Wesley Publishing Company, 1990.

[6] D'Azzo & Houpis. Linear Control System Analysis and Design. Mc Graw-Hill, Inc. 1975.

[7] Archibald Joseph Macintyre. Máquinas motrizes hidráulocas. Guanabara Dois, 1983.

[8] Karl J. Aström & Björn Wittenmark. Computer-controlled systems, theory and design. Dover Publications, Inc. 1997.

[9] Saber Elaydi. An introduction to difference equations. Springer. 2005.

[10] Arduino. www.arduino.cc 2013.

[11] Michael Margolis. Arduino Cookbook. O'Reilly Media Inc. 2011.

[12] Ramakant A. Gayakwad. Op-Amps and Linear Integrated Circuits. Prentice Hall. 1993.

[13] Texas Instrument www.ti.com datasheet op. amp. 741. 2013.

[14] Jacob Millman and Christos C. Halkias. Integrated Electronics. Mc Graw-Hill, 1972.

[15] João Marcos Castro Soares. Identificação experimental, simulação e ajuste de reguladores de velocidade de usinas hidrelétricas com verificação em testes de campo. Dissertação de mestrado. Universidade Federal de Santa Maria – UFSM, Santa Maria-RS. Brasil. 1982.