A DESIGN OF THE PID BASED TEMPERATURE CONTROLLER

Aleksandar Žorić
Faculty of technical sciences
University of Priština, Serbia

Aleksandar Micić
Faculty of technical sciences
University of Priština, Serbia

Verica Vasiljević
Faculty of information technologies
Slobomir P University, R. Srpska-BiH

Nikola Žorić
Faculty of technical sciences
University of Priština, Serbia

Abstract

This paper describes a practical solution of PID based temperature controller. The discrete PID algorithm was implemented by software using a 32-bit microcontroller. As the heater driver, phase controlled triac device was used, while the DS18B20 is selected as a digital temperature sensor. The realized temperature controller was tested on a real process. The test results show good agreement with theoretical assumptions.

Proposed controller is designed to be implemented into heating system which can be represented as FOPDT (First Order Plus Dead Time) model. The cost of the implemented system is relatively small in relation to the opportunities and savings that can be achieved by its implementation.

Keywords: PID, FOPDT, PLX-DAQ, SSR temperature controller, MCU, triac, Ziegler-Nichols method.

INTRODUCTION

PID (*Proportional Integral Derivative*) controllers are the most represented regulators in the industry with more of 95% participation of all of used regulators [1], [7]. The development of fast and powerful microprocessors and microcontrollers made it possible to implement digital PID controllers in the form of standalone devices or as part of a PLC (*Programmable Logic Controller*). Today's modern industries, scientific workstations, laboratories, and the like use the control systems whose extensive application is also seen in regulating vehicle speed, temperature control of incubators for premature infants, temperature and humidity control of cellular incubators, mobile robot control and many other applications.

The PID algorithm is a relatively easy to conceptually understand and implement in practice [3]. The favorable cost/benefit ratio provided by PID controllers make them the most commonly used controllers in the lowest level of control systems in industry.

In this article the PID based temperature controller is proposed. Simple architecture and low cost are the main features of the proposed controller. Due to the application of the semiconductor temperature sensor, the operating temperature range of the controller is limited from -55 °C to 125 °C. This deficiency can be dispersed by a slight modification of the sensor circuit of the proposed solution.

CONTROLLER ARCHITECTURE

The block schematic of the controller represents the conventional regulation circuit shown in the Figure 1.

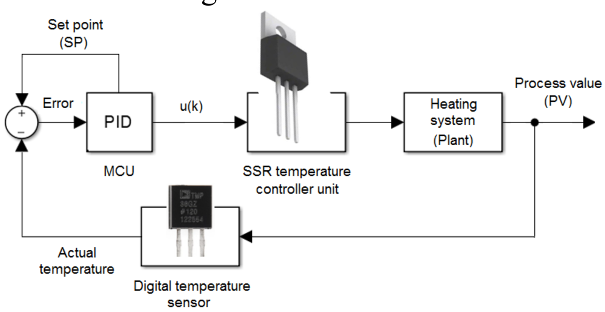

Fig. 1. *PID temperature controller block schematic*

As can be seen, the output of the PID regulator drives the SSR temperature controller unit [5] while the feedback makes the digital temperature sensor DS18B20. The PID regulator was implemented as software code using the microcontroller. The heating system can be represented as first order system plus dead time model (FOPDT) [1], [7]. Such models can represent various plants in the industry.

In the Figure 2 complete schematic of the proposed PID based temperature controller is reported.

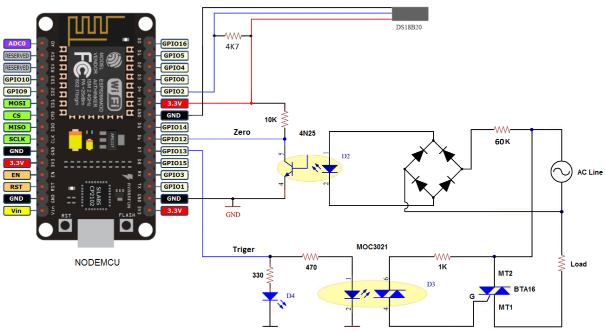

Fig. 2. *PID temperature controller complete schematic*

The heart of the controller is the NodeMCU platform with 32-bit microcontroller (MCU) *Tensilica's L106 Diamond* with low power consumption and maximum clock frequency of 160MHz [4]. Primarily, the development platform was created on the basis of the ESP8266EX SoC (*System on a Chip*) with a high degree of integration with the MCU that can host the application or work as a slave with an external host MCU. Integrated serial communication interfaces, such as SPI (*Serial Peripheral Interface*), I^2C (*Inter-Integrated Circuit*), or UART (*Universal Asynchronous Receiver-Transmitter*), can be a convenient coupling when using the ESP8266EX SoC as a WiFi adapter in any microcontroller environment. When the ESP8266EX SoC hosts an application, it then "boots up" from the fast external flash memory. SoC integrated fast cache is then of direct help to increase system performance and optimize system memory. In accord with this, the ESP8266EX SoC contains integrated 50kB SRAM (*Static Random-Access Memory*) and ROM (*Read-Only Memory*) units as well as a memory controller. The 17 GPIO lines of SoC can be used as an interface with external sensors and other devices.

Temperature controller unit contains opto-isolated zero-cross circuit to detect the passing of AC (*Alternating Current*) voltage across null, as well as the power triac controlled by optically coupled triac driver [5]. Each passage of AC voltage through zero generates an interrupt signal at the GPIO12 (*General-Purpose Input/Output*) input line of the MCU. Based on the output control signal of the PID regulator, the MCU calculates the ignition angle of the triac and switches on the triac with a short pulse on the GPIO13 output line. By change of ignition angle of the triac in each half period of AC voltage the effective value of the voltage on the load can be controlled. In such a way, the active power of the heater of the heating system can vary in accord with the value of the output variable of PID regulator.

As a feedback of the controller the digital temperature sensor DS18B20 was used. Selected temperature sensor has user configurable temperature resolution 9, 10, 11 and 12 bits, corresponding to temperature change of 0.5°C, 0.25°C, 0.125°C, and 0.0625°C, respectively. The DS18B20 communicates over a 1-Wire bus that requires only one data line for communication with a MCU and an external pull-up resistor. Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20's to function on the same 1-Wire bus. Operating temperature range is from -55°C to 125°C. Serious disadvantage of the DS18B20 sensor is a long temperature conversion time of max 750ms. However, having in mind the inertia of the heating system and possibility of non-blocking temperature measurement using selected sensor, this global disadvantage will not has influence on the sampling rate of PID regulator. Finally, the operating temperature range of the applied sensor remains as a limitation.

HEATING SYSTEM MODELING

Proposed controller was tested on a heating system consisting of the filament of a light bulb for AC line. The light bulb of 15W power for 220V and 50Hz was selected. The temperature sensor was mounted at the glass balloon surface of the lamp. Such a choice of the heating system was made because of the shorter duration of the experiments, in accord with relatively small time constant of the system.

One of the well known experimental methods for PID parameters tuning is the Ziegler-Nichols's step excitation of the plant in open loop (without regulation) [3], [1]. It works quite well on a large number of the first-order systems and does not require extensive system testing or measurement. Assuming that the selected heating system can be modeled as first-order system with a transport delay (FOPDT) it was necessary to measure the temperature response of the plant on the step change of the input heating power and then modeling the plant.

Figure 3 shows the step response of the heating system in open loop.

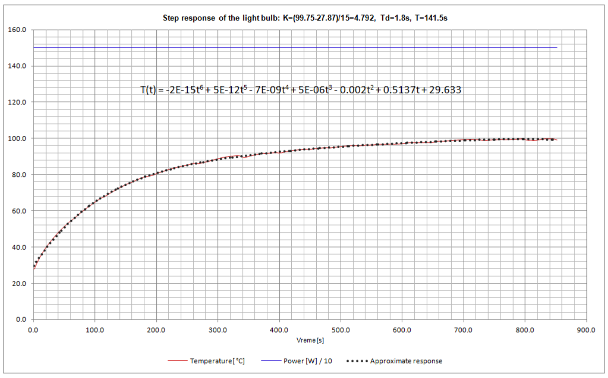

Fig. 3. *Plant open loop temperature response and approximate response on the step change of the input heating power*

The displayed system response was recorded in real time using PLX DAQ v2 program [6]. PLX DAQ v2 is a program used to establish an easy communication between Microsoft Excel on a computer with Windows operating system and any device that supports serial port protocol.

As can be seen in the Figure 3, the step response of the plant corresponds to the response of FOPDT system with transfer function in the form

$$G(s) = \frac{K}{Ts+1} e^{-T_d s} \qquad (1)$$

where K is the gain, T is time constant and T_d is dead time of the plant.

The follow model parameter values were obtained from the experiment, i.e. from Figure 3:

- dead time - $T_d=1.8s$,
- step input (power input to the plant) - *15W*,
- steady-state change in output - *(99.75-27.87)* $^{\circ}C = 71.88\,^{\circ}C$.
- gain - $K = \frac{71.88^{o}C}{15W} = 4.792 \left[{}^{o}C/W\right]$,
- the temperature corresponding to the time constant of the plant is $0.632(71.88)\,^{\circ}C + 27.87\,^{\circ}C \approx 73.3\,^{\circ}C$

Thus, from Figure 3, the time constant is: *T=(143.3 - 1.8)s=141.5s.*

Then the equation (1) becomes,

$$G(s) = \frac{4.792}{141.5s+1} e^{-1.8s} \qquad (2)$$

The basic advantage of the described methodology of obtaining a model (process identification) lies in its simplicity. By measuring only three parameters (K, T and T_d), the model of the process is determined within one experimental test.

PID PARAMETERS TUNING

As it was known, PID control signal is the sum of three terms which are based on the error signal:

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \qquad (3)$$

where $e(t)=r(t)-y(t)$ represents error as the difference of reference and output values, K_p is proportional constant, K_i is integral constant and K_d is derivative constant. By discretization of the equation (3) in the time domain, obtained expression was implemented as difference equation applicable for digital control systems as microprocessor implementation.

Regulator	K_p	T_{int}	T_{der}
P	$1/(K\Theta)$	-	-
PI	$0.9/(K\Theta)$	$3T_d$	-
PID (parallel)	$1.2/(K\Theta)$	$2T_d$	$T_d/2$

***Table 1**. PID parameters choice, determined by Ziegler and Nichols for step excitation of the plant in open loop*

The well chosen value of PID parameters K_p, K_i and K_d are the key in providing stable and desired transient and steady-state response. PID parameters K_p, K_i and K_d can be obtained by using heuristic methods, analytical methods, frequency response methods, optimization methods and adaptive tuning methods [1], [2], [3]. The values of K_p, K_i, and K_d, are obtained here as proposed by Ziegler and Nichols [3] - Table 1, where $\Theta=T_d/T$ is normalized time, $K_i=K_p/T_{int}$ and $K_d=K_pT_{der}$.

NODEMCU PROGRAMMING

The main task of the selected 32-bit NodeMCU platform [4] is the calculating the PID controller's control variable based on the error signal (difference between reference and controlled temperature) at specified time intervals and accordingly driving the SSR (*Solid State Relay*) temperature controller. This task can be accomplished by constantly monitoring the controlled output temperature of the plant. The complete task was solved programmatically using the popular Arduino IDE (*Integrated Development Environment*).

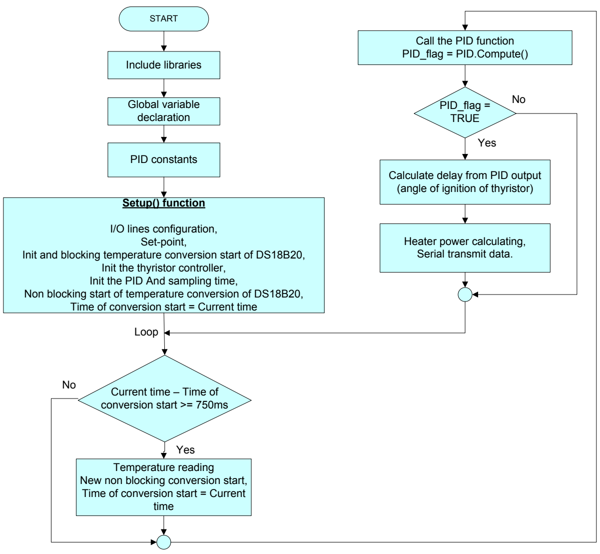

Fig. 4. *Main program flow diagram*

As can be seen from Figure 4, the main loop executes by cyclic with periodic asynchronous (non-blocking) 12-bits temperature reading. Within the loop, a *PID.Compute()* function is called that calculates output only at the appropriate sampling time. The PID regulator output control variable is an unsigned eight bits integer number *N*. The delay calculated on the basis of this number is given by the expression (4). The delay decreases with increasing *N*.

$$t_d = 10000 - N\,10000/255 \;\; [\mu s] \qquad (4)$$

The calculated delay is used in the interrupt routine to switch on the thyristor at the appropriate ignition angle in each half-period of AC voltage of frequency 50Hz. The interrupt routine is called each time the AC voltage goes through. The serial data transmission format is adapted to the mentioned PC's program for acquisition and monitoring - PLX DAQ v2.

RESULTS AND DISCUSSION

Figures 5 and 6 show the waveforms of the responses of the tested controller - temperature on the glass surface of the light bulb and electric power of the light bulb, recorded in real time with PI and PID regulators, respectively. The constants of one and the other type of regulator were calculated on the basis of the parameters of the recorded model of the plant and Table 1. A dedicated PLX DAQ v2 program [6] was used to observe and record the waveforms. Beside the controlled output temperature, the controlled electric power of the light bulb was also observed and recorded in order to gain a quantitatively better insight into the regulation process. The responses of the controller without disturbances were recorded.

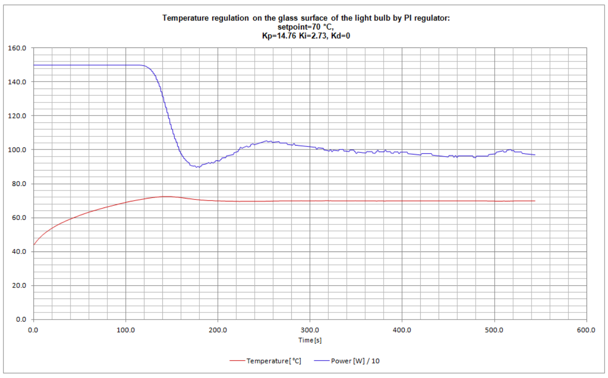

Fig. 5. *The responses of temperature and active power when using PI regulator at setpoint 70 °C*

Looking at the recorded waveforms, it could be said that the control with the PID regulator is finer - more accurate than the PI regulator. Although the starting and reference temperatures are different in both cases, it is also noticeable that the settling time is shorter in the case of PID control.

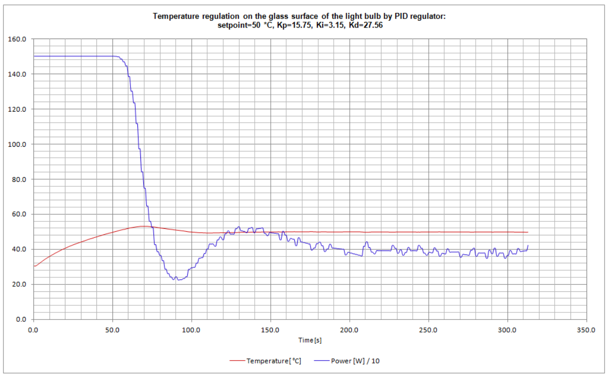

Fig. 6. *The responses of temperature and active power when using PID regulator at setpoint 50 °C*

Figures 7 and 8 show the effects of external disturbance on PI and PID regulators with the same constants as in the previous two cases. The external disturbance was caused by the forced cooling of the glass bulb of the lamp.

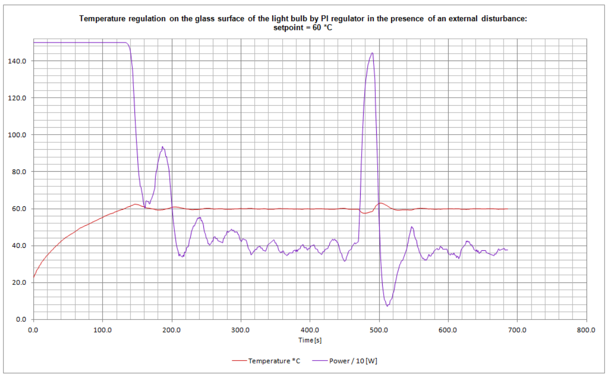

Fig. 7. *The responses of temperature and active power of PI regulator with external disturbance*

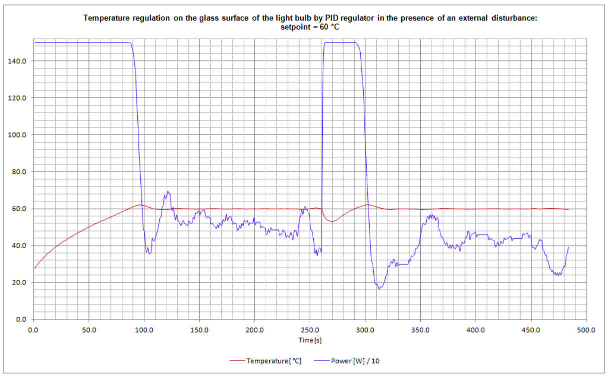

Fig. 8. *The responses of temperature and active power of PID regulator with external disturbance*

In the case of short-term external disturbance, a relatively fast stabilization of the controller to the reference - setpoint temperature was observed in both cases with the PI and PID regulator.

Finally, the effects of the disturbance caused by the change in the setpoint temperature on the controller with the PI and PID regulator are illustrated in Figures 9 and 10, respectively.

The reference temperature changes was made during the acquisition and monitoring process by programmatically reading the setpoint temperature from the corresponding PLX DAQ user interface table field. As can be seen from Figures 8 and 9, in both cases the controller, after reaching the new reference temperature and the transition mode, stabilizes the output temperature to the new setpoint, whether it is a positive or negative change in the reference temperature.

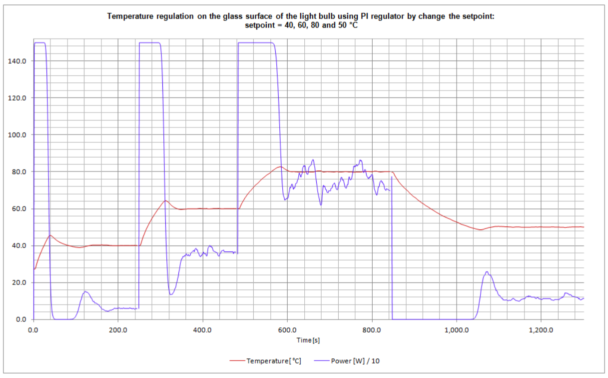

Fig. 9. *The responses of temperature and active power of PI regulator with disturbances caused by change the setpoint temperature*

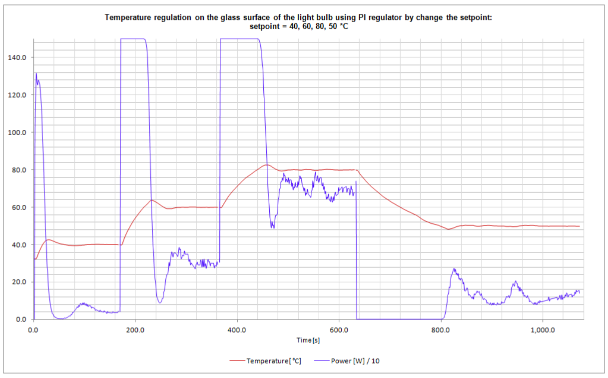

Fig. 10. *The responses of temperature and active power of PID regulator with disturbances caused by change the setpoint temperature*

CONCLUSION

The main objective of this work was the design and implementation of a modern, digital control system with PID controller for precise temperature control of the respective control object. According to the price/

performance ratio, the realized system belongs to the group of optimal systems of low cost and solid performance. The system has been tested in several iterations and under different operating conditions (with and without disturbance) has not shown defects that could be assumed, such as: entry into saturation of integral controller member, inability to enter after-disturbance control mode, reliability, etc. Due to the large and diverse offer of integrated components of the described system and services in the market, the price of the realized system is relatively small in relation to the possibilities and savings that can be achieved by its implementation.

The proposed control system can be installed in any electrical heating device whose response to the step excitation corresponds to a first order system with a transport delay model. In conjunction with a PC, it is relatively easy to adjust the PID controller parameters and adjust the system to the control object.

Further qualitative work on the development of the described system could go towards setting up the PID controller via the appropriate Web interface, with regard to the selected microcontroller platform (SoC) with an integrated WiFi interface. What's more, the actual acquisition and monitoring process could be done wirelessly on the basis of a dedicated Web application. Such an approach would allow more adjusting and monitoring of the implemented control system. Also, the selected hardware and software unit offers the ability to implement far more complex management laws.

ACKNOWLEDGEMENT

This work was supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia within the project III 47016.

REFERENCE

[1] Trafczynski, M., Markowski, M., Alabrudzinski, S. and Urbaniec, K., “Tuning Parameters of PID Controllers for the Operation of Heat Exchangers under Fouling Conditions”, *Chemical Engineering Trans*., Vol. 52, 2016.

[2] Marzoughi, A., Selamat, H., Rahmat, M., F., and Rahim, H. A., “Optimized proportional integral derivative (PID) controller for the exhaust temperature control of a gas turbine system using particle swarm optimization”, *Int. Journal of the Physical Sciences* Vol. 7(5), pp. 720-729, 30 Jan., 2012.

[3] Salim, Kumar, S., Ohri, J., “LabVIEW Based DC Motor and Temperature Control using PID Controller”, *Int. Journal of Advanced Research in Computer Science and Software Engineering*, Vol. 3, Issue 5, pp 56, May 2013.

[4] ESP8266 NodeMCU WiFi Devkit, Available: *www.handsontec.com/pdf_learn/esp8266-V10.pdf*

[5] The Basics of SSRs: The Switching Device, Available: *https://www.allaboutcircuits.com/technical-articles/basics-of-ssr-solid-state-relay-the-switching-device/*

[6] Beginners Guide to PLX DAQ v2, Available: *https://www.academia.edu/32577862/Beginners Guide-to_PLX_DAQ_v2_by_Net_Devil*.

[7] Kalaiselvan, E. and Tagore, J. D., “A Comparative Novel Method of Tuning Controller for Temperature Process”, *Int. Journal of Advanced Research in Electrical, Electronics and Instrum. Engineering*, Vol. 2, Issue 11, November 2013.