Advanced Research in Electrical and Electronic Engineering
p-ISSN: 2349-5804; e-ISSN: 2349-5812 Volume 6, Issue 2 April-June, 2019, pp. 82-86

http://www.krishisanskriti.org/Publication.html

Automatic Mixing of Water using a Feedback Controlled System with Face Recognition Feature

Mohmad Saleem Mir*¹, Kamran Iqbal Naqash*², Adam Nawaz*³,
Junaid Muzaffar*⁴ and Dr. Bilal Ahmad Malik#

**B.Tech Student, Department of Electronics & Communication Engineering,*
Institute of Technology, University of Kashmir, Zakura Campus
#Scientific Officer, Department of Electronics, IOT, University of Kashmir
E-mail: ¹mirsaleem2626@gmail.com, ²kamraaniqbal222@gmail.com,
³adamnawaz@live.com, ⁴bilalmalik@kashmiruniversity.ac.in

Abstract—The proposed method is a novel low cost solution of mixing hot and cold water automatically to a desired temperature. The traditional method makes use of manual water mixer, which is not precise and takes a number of attempts to get the desired temperature. This has safety issues, as it can cause scalding, if temperature of mixed water is above 48℃. This can be very dangerous for children. The proposed method uses a feedback controlled system in which Raspberry Pi measures the temperature of mixed water continuously to control the position of hot and cold water valves to get the water with desired temperature. Since different people have different perception of temperature, the temperature can be set either through a keypad or making our system to predict the temperature for family members automatically by Face Recognition using Machine Learning. This provides seamless experience as we do not even need to enter the temperature through keypad every time. Rather we need to train our machine once. The camera sensor on Raspberry Pi can then identify the members and thereby making everyday use effortless.

1. Introduction

Mixing hot and cold water is a herculean process. It takes a number of attempts to mix water to a desired temperature. Even after so many attempts, the output temperature is not precise. Also, in attempting to get the water with desired temperature, our skin is frequently scalded, when it comes in contact with water having greater than 48°C. This can even cause burn injuries. The American Burn Association states that a scalding injury can occur in one second when skin is placed in contact with water measuring 68°C[1]. The hot tap water scalding injuries account for 5,000 child scalding injuries each year[2]. These problems are solved by the proposed method that can mix the hot and cold water automatically to a desired temperature for safe use.

The system works around Raspberry Pi, being used as a controller for valves. To design a low cost electrically controllable valve, we attached servo motor to a manual valve. In order to have a precise control, we designed a PID controller for valve, simulated it and then implemented it on the prototype. Since different people have different perception of temperature, the temperature can be set either through a keypad or making our mixer predict the temperature for family members automatically by Face Recognition. The faces of family members along with their labels (desired temperatures) are fed to machine and trained with some machine learning algorithm. Next time when it picks a face, it automatically sets the temperature using face recognition. This provides seamless experience as we do not even need to enter the temperature through keypad every time. Rather we need to train our machine once. The camera sensor on Raspberry Pi can then identify the members and thereby making everyday use effortless.

2. Block Diagram

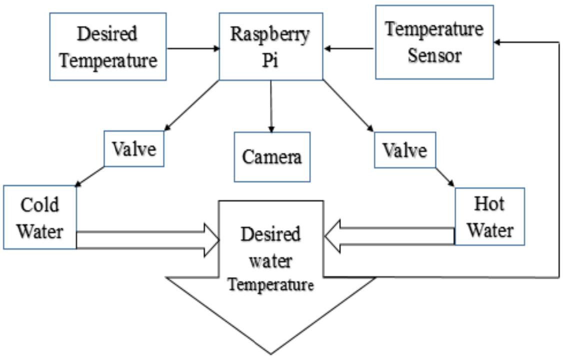

Fig. 1: Block Diagram of Smart Water Mixer

3.1. Measuring Temperature using DS18B20

In this project, we have used DS18B20 temperature sensor for temperature measurement. The DS18B20 digital thermometer provides 9-bit to 12-bit Celsius temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18B20 communicates over a 1-Wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. In addition, the DS18B20 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply. Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20s to function on the same 1-Wire bus. Thus, it is simple to use one microprocessor to control many DS18B20s distributed over a large area [3].

3.2. Benefits and Features of DS18B20

— It has aUnique 1-Wire Interface that requires only one Port Pin for Communication.

—Reduced component count with Integrated Temperature Sensor and EEPROM.

—Measures Temperatures from -55°C to +125°C (-67°F to +257°F).

— ±0.5°C Accuracy from -10°C to +85°C.

— Programmable Resolution from 9 Bits to 12 Bits.

— No external components required.

— Parasitic Power Mode requires only 2 Pins for operation (DQ and GND).

— Flexible User-Definable Nonvolatile (NV) Alarm settings with Alarm Search Command, which identifies Devices with Temperatures outside Programmed Limits.

— Available in 8-Pin SO (150 mils), 8-Pin µSOP, and 3-Pin TO-92 Packages. [1]

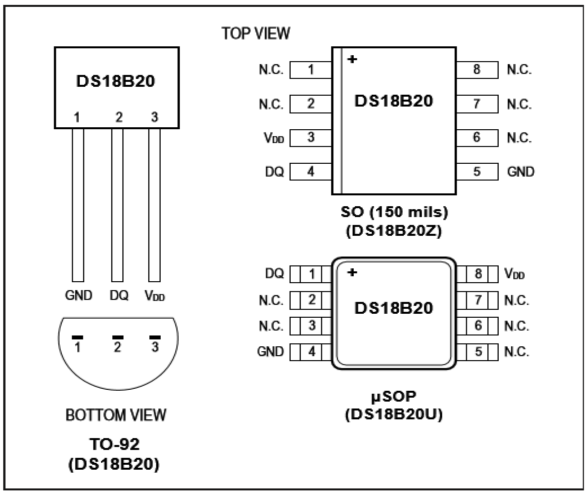

Fig. 2: DS18B20 Pin Configurations [3]

A typical error curve for DS18B20 is shown as follows:

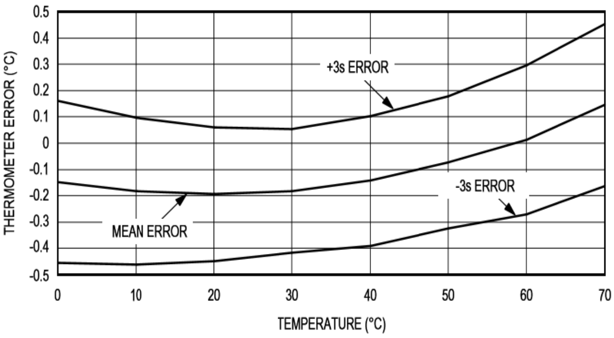

Fig. 3: DS18B20 Typical error Curve [3]

As follows from the above curve, it is optimum to use DS18B20 for our proposed method of water mixing. Since hot water is usually stored nearly at a temperature of 60-70℃, the error will be small.

4. Controlling Valves

In a closed loop (feedback) control system, the output is fed back and compared to the command input (the difference between the two is called ass error) to see how far our system from the target[4]. If the output is exactly equal to the target value, the error will be zero as required.

We started simulation with a P Controller in Scilab as shown below:

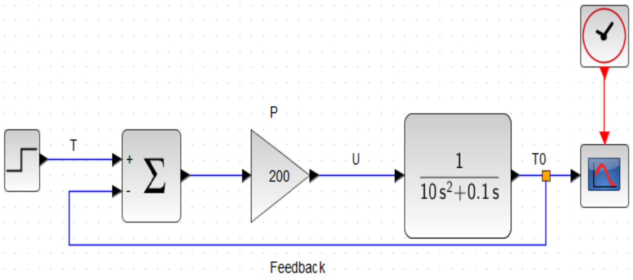

Fig. 4: Servo motor with Proportional controller

T: Desired Temperature

T0: Output Temperature

u: Actuating Signal (Input to servo controlled valve)

$u = kp*e = kp*(T-T0)$……………….1

The problem with proportional controller is that as time progresses, error decreases, which causes decrease in actuating command (u) and hence position of valve will start decreasing. Once we reach our target, i.e.,

$T = T0$ or $u = 0$ ……………………..2

When actuating command becomes zero, this will bring valve back to its original position i.e., valve will get closed.

The step response of valve with a proportional controller is shown as below:

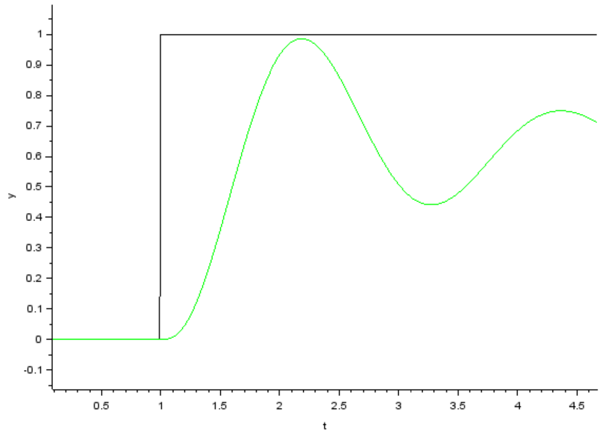

Fig. 5: Step response of servo motor with a proportional controller

As Valve closes, the error will again start increasing. This will again generate a non-zero actuating signal and valve will open again. If the gain is very large, the system will start oscillating. So, we need to modify our P controller, such that even when error is zero, the output of the controller should not be zero and once the error becomes zero, the output of the controller should remain constant. This can be done by letting our controller use the past information i.e., by adding an integral path in our controller as shown below:

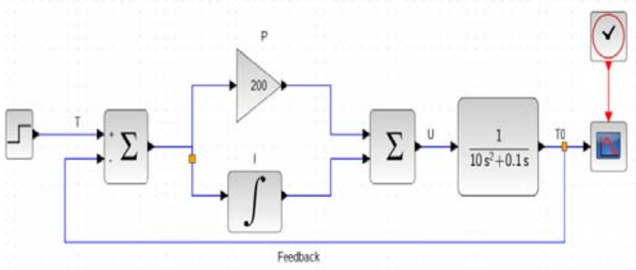

Fig. 6: Servo motor with PI Controller

An Integrator sums up the input values over time, keeping a running total. Therefore, it has that memory of what has happened before [4].Basically, it keeps the track of the Past. Now, as long as there is a positive error (T>T0) in our system, the integral output will continue increase. This increased value from integral path will increase the valve position and thus output temperature will keep on increasing as desired.

When T0 = T, error = 0……………..3

the output of proportional path will become zero, but the output of integral path will be non-zero and it will be the value required for the valve to operate as desired. for valve to operate properly. The step response of servo motor with a PI Controller is shown as follows:

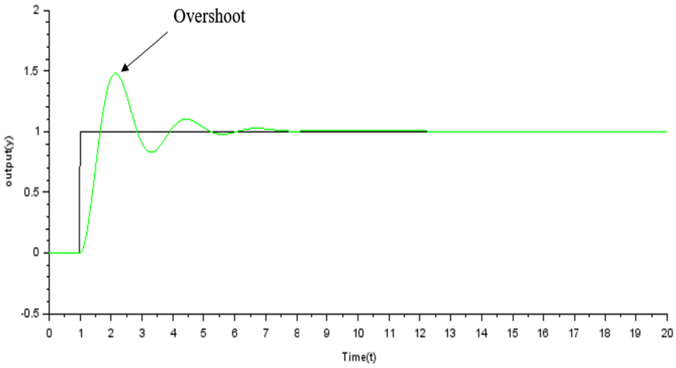

Fig. 7: Step response of Servo with a PI Controller

These two paths, the integral and proportional path work with each other to drive error to zero and at the same time maintaining a constant actuator command. But, there is a problem with PI controller as illustrated below:

Let desired temperature be 35℃ for which we assume that the angular position of hot water valve should be 70º. Just before our system reaches the 35℃, the proportional path is basically zero, since the error is small. But, the integrator may have summed to a value over 70º angular position. This will cause the temperature to keep rising, since we are still below 35℃. However to remove that excess angular position, the output temperature will have to go > 35℃ to create a negative error. This negative error when summed, lowers the output of integrator and reduces the angular position of valve. However, this overshooting might not be desired. For example, if the desired temperature is 48℃, but because of overshoot, it raised to greater than 50℃. This can lead to scalding.

This problem can be removed by adding a path to our controller that can predict the future and respond to how fast we are approaching our goal [7]. This can be done by a derivative as shown below:

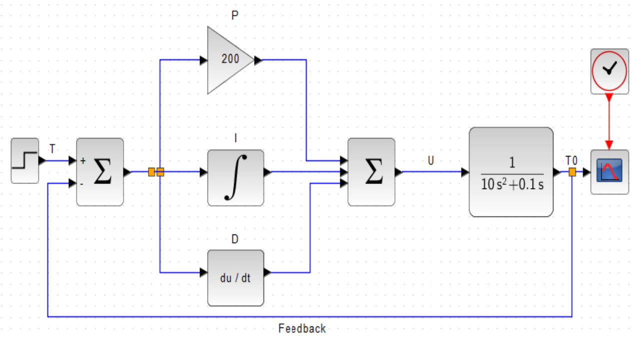

Fig. 8: Servo motor with PID Controller

A Derivative produces the measure of rate of change of error i.e., how fast the error is growing or shrinking. Hence if the angular position of valve is changing quickly, the error will decrease quickly [5].That decreasing error has a negative rate of change and thus derivative will produce negative rate of change, decreasing the angular position of the valve and thus compensates for the overshoot. This is illustrated in following graph:

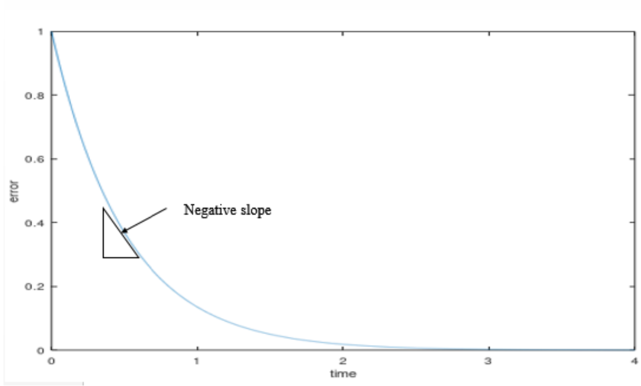

Fig. 9: Change in Error vs time

Basically, our controller is using the changing error to determine that we are approaching our goal way too fast and then prematurely reducing angular position of valve, preventing system from overshooting.

Another advantage of using Derivative path is illustrated as follows:

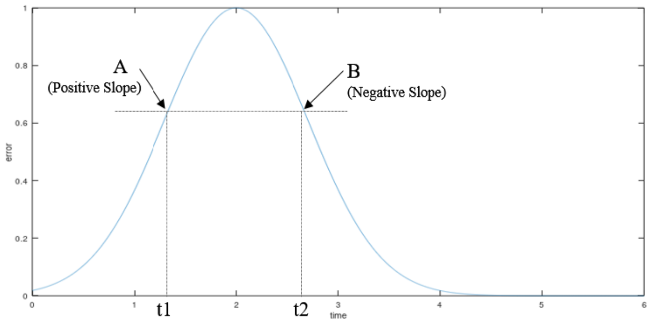

Fig. 10: Variation of error vs. time

Let us assume that the variation of error with time is as shown above. Let Desired temperature = 35℃, for which we assume 70° angular position of hot water valve.At point A in the graph, the error is increasing. In order to reduce the error, the angular position of hot water valve should increase. But at point B, angular position should decrease, since the error is decreasing. An PI controller cannot differentiate between the two points, since the value of error at both the points is same. However, a Derivative path measures rate of change of error, which will be positive at point A and negative at point B, thereby increasing the angular position at A and decreasing at B.

By using a PID controller, our system works as desired. To get an optimum PID control, we have tuned our PID controller to get the proper values for Proportional, Integral and Derivative gain as shown below:

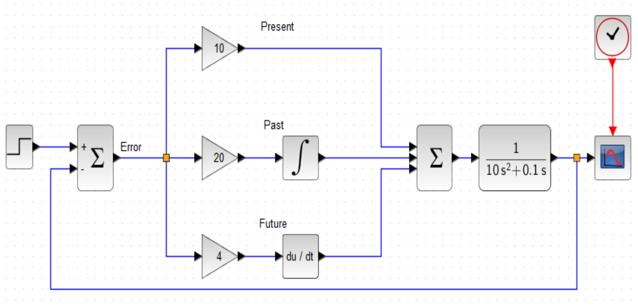

Fig. 11: PID Controller with proper values of gains

5.1. Face Recognition

Although facial recognition sounds like a very simple task for us, it has proven to be a complex task for a computer, as it has many variables that can impair the accuracy of the methods, for example: illumination variation, low resolution and occlusion, amongst other. In computer science, face recognition is basically the task of recognizing a person based on its facial image. It has become very popular in the last two decades, mainly because of the new methods developed and the high quality of the current videos/cameras. With the facial images already extracted, cropped, resized and usually converted to grayscale, the face recognition algorithm is responsible for finding characteristics which best describe the image [6].

The face recognition systems can operate basically in two modes:

Verification of a facial image: It basically compares the input facial image with the facial image related to the user which is requiring the authentication. It is basically a 1x1 comparison.

Facial recognition: It basically compares the input facial image with all facial images from a dataset with the aim to find the user that matches that face. It is basically a 1xN comparison.

There are different types of algorithms for facial recognition. One of the easiest and efficient algorithm was given in the year 1996. It is known as **Local Binary Patterns Histograms (LBPH)[8].**

5.2 Local Binary Pattern Histogram

This method takes a window (say 3x3) and moves it all over the face. At each move, it compares the intensity of neighboring pixels with center pixel. If the intensity of neighbouring pixel is greater than central pixel, a value of 0 is assigned, otherwise 1 is assigned. Now the binary pattern is read at each move and converted to the decimal values. Then a histogram of these values is computed. Thus algorithm remembers the histogram of a face along with its label. Next time when a new image is fed, it computes its histogram and then compares with known histograms to fit the best match[8].

6. Conclusion

The proposed method provides us with the efficient low cost design of the water mixer that can help the general masses. It is a new method for safe use of hot water for household use, especially for children.

In the paper we have designed a feedback regulated system for mixing of cold and hot water. This method can be further extended for mixing of chemicals in a given proportion in the Lab, but instead of temperature sensing, we have to sense the concentration of the chemical.

Limitations

1. Although servo controlled proportional valve are less costly, but they are less precise than the pneumatic valves.
2. Face Recognition has a delay of 5 to 10 seconds which can be removed by using a better face recognition algorithm and better hardware.

References

[1] Moritz, AR, Herriques, FC Jr. Studies of thermal injuries: II The relative importance of time and surface temperature in the causation of cutaneous burns. Am J Pathol 1947; 23:695-720.

[2] National Safe Kids Campaign. Facts About Childhood Burns. 2006

[3] Maxim Integrated Products, "Temperature Sensors", DS18B20 datasheet, 2018

[4] Karl Johan Astrom, "Control System Design", 2002

[5] I.J. Nagrath and M. Gopal, "Control Systems Engineering", New Age International Pvt. Ltd Publishers, December 1, 2005

[6] Proyecto Fin de Carrera, "Face Recognition Algorithm", June 16, 2010

[7] Farid Golnaraghi and Benjamin C. Kuo, "Automatic Control Systems", Wiley, 1962

[8] J. Zou, Q. Ji, and G. Nagy, "A comparative study of local matching approach for face recognition," Image Processing, IEEE Transactions on, vol. 16, no. 10, pp. 2617–2628, 2007.