Simulator for Water Dispenser

Ray-Guang Cheng
Department of Electronic and Computer Engineering,
National Taiwan University of Science and Technology
Taipei, Taiwan, Republic of China
Email: crg@ieee.org

Ankit Kumar Panda
Department of Electronics and Communication Engineering,
National Institute of Technology,
Rourkela, India

***Index Terms*—Simulator, C++, Water Dispenser**

I. System Model

This paper presents the design of a simulator for a drinking water dispenser deployed in [1]. The functional blocks and a basic functioning of the simulator is as shown in Fig. I.

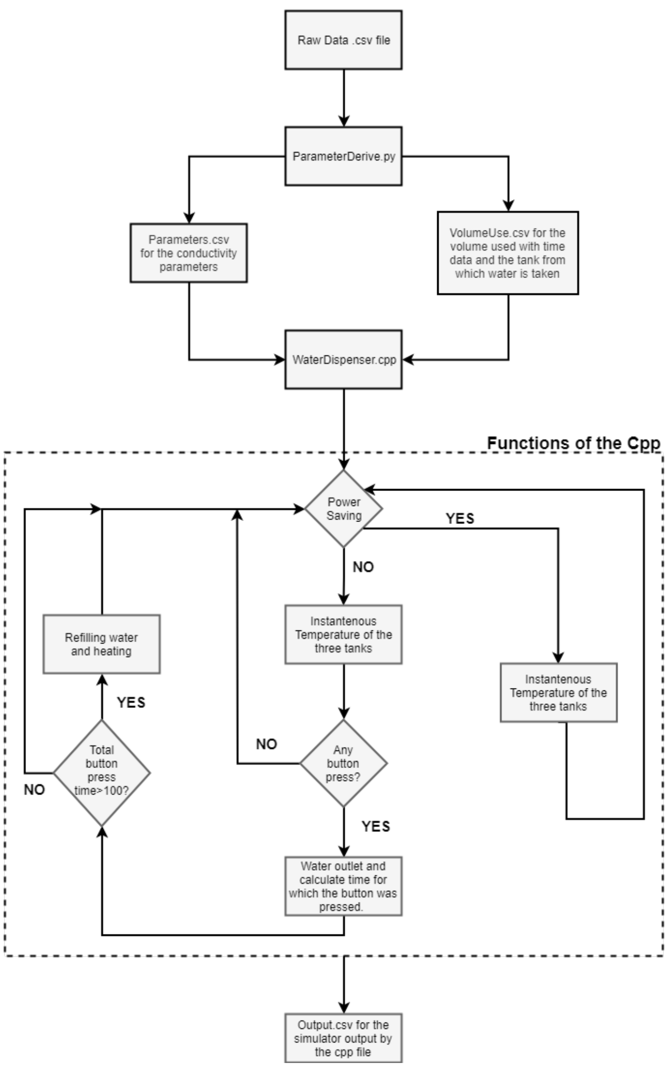

Fig. 1 Functional modules of the simulator.

This work was supported in part by the Ministry of Science and Technology under Contract 105-2221-E-011-033-MY3 and 106-2218-E-011-007-MY2.

The configuration parameters of the dispensers are defined below. Let $T_{H,u}$/$T_{H,l}$ and $T_{C,u}$/$T_{C,l}$ be the upper/lower thresholds of the water temperature in hot and cold tanks, respectively. The dispenser turns on the heater as soon as the water temperature drops to $T_{H,l}$ and the heater remains on till the water temperature reaches $T_{H,u}$. Similarly, dispenser turns on the chiller as soon as the water temperature raises to $T_{C,u}$ and the chiller remains on till the water temperature drops to $T_{C,l}$.

Let $T_{H,ini}$, $T_{W,ini}$, and $T_{C,ini}$ be the initial water temperature of the hot, warm, and cold tanks, respectively. Let k_H and k_C be the *thermal* conductivity of the insulator around the hot and cold tanks, respectively.

Let T_H, T_W, and T_C be the instantaneous water temperatures of the hot, warm and cold tanks, respectively. They are the output parameters of the simulator.

The dispenser simulator consists of the following functional modules:

1) Power Saver The module consists of two strings which can be changed to get the different power saving timings. If the value of the power saving flag is true, the simulator runs in the power saving mode operation.
2) Input parameters The module reads the parameters given by the parameters.py. The parameters.py takes the raw data and processes it to obtain the parameters into a csv file. The csv file is read by the module and is passed on to the other modules.
3) Hot tank refill This is the module working for the refilling of the dispenser. It consists of the temperature drop happening in the hot tank when water is refilling. Also, it sets the refilling flag to true during the refilling process.
4) Instantaneous temperature hot/warm/cold The value of the temperature at the instant on any of the three tanks is given by this module. The modules also give the temperature during the heating, cooling and refilling process for the hot, cold and warm tank respectively. These functions pass the values to WriteFile module which saves the temperature data.
5) Tank GUI This module is used for generating the display GUI of the volume and temperature of each tank. Additionally it also detects the button pressing simulation and sends the command to the button press module. It also calculates the total time for which all the tank buttons

are pressed, if greater than 100 it sends command to hot tank refilling module.

6) Button Press Hot/Warm/Cold The module is used to simulate the button pressing by the user. This function updates the volume used in total by the simulator and updates the current volume of water available in each tank. This module is interconnected with the Tank GUI module.
7) Write File This module is used to save the data of the temperature, volume used and the flag statuses for the dispenser to a csv file. This module implements the historical data saving option for the simulator.

The fresh water temperature refilling into the hot tank is denoted as FW_T. The refilling rate of the dispenser is denoted by the symbol R_t. The individual tank water outlet rate can be defined as H_t, W_t, C_t for the hot, warm and cold tank respectively. F_r be the flag for refilling status, F_h be the flag for the heating status, F_c denote the flag for the cooling status, F_p be the flag for the power saving mode status. The amount of time for which all the three buttons are pressed after which the refilling will take place be denoted by r_t. Let V_h, V_w, V_c be the volume of water consumed in the hot, warm and cold tank during a minute interval. Let V_t be the total volume of water consumed by the dispenser. The work uses Linear regression techniques hence, the statistical parameter R^2 is used to check the goodness of the fit of the curve.

The parameters derived from the collected data are listed below. The range of the temperatures for the hot (UH_T - LH_T) and the cold tank (UC_T - LC_T) are taken as an input from the user. The warm starting temperature (SW_T) and the filling water temperature (FW_T) is also taken as an input from the user. The calculation of the conductivity values for the tanks during the normal heat loss is done by using the linear regression tool of microsoft excel. From the raw data the range of temperature of the hot tank is selected when the water is not taken out and F_h is 0, the equation is then derived by fitting these points from the collected data by a linear line as shown in the figure 2. The slope of the graph gives us the value of the conductivity of the hot tank. The intercept of the line is the value given by LH_T.From the figure 2 we see that value of k_H is -0.047.

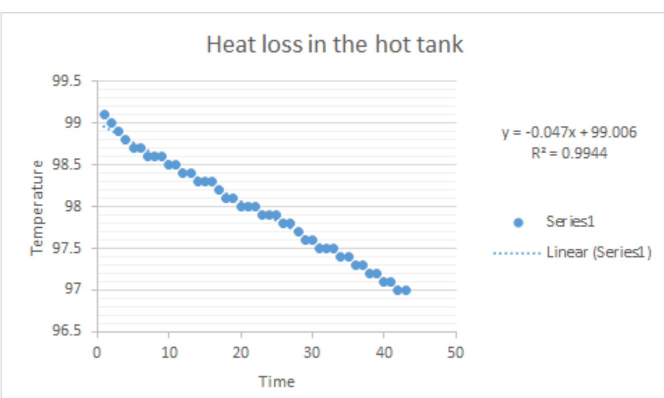

Figure 2 shows the temperature against time for the hot tank.

To derive the equation for the heating of the water in the hot tank, the temperature range is selected from where the value of F_h is 1 and the linear regression is applied for the hot temperature tank range and the result can be seen in the figure 3. The figure gives us the equation for the heating process of the tank. The intercept of the line is the value given by LH_T.

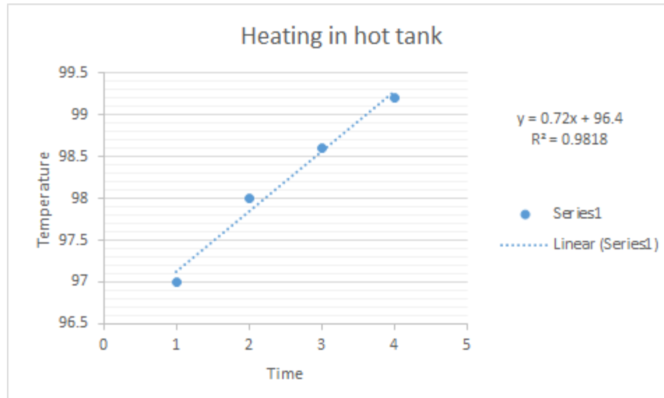

Figure 3 shows the heating process for the hot tank.

Similarly, for the warm tank, there is a heat loss in the tank over time, the conductivity in this case when F_r is 0 can be determined from the figure 4.

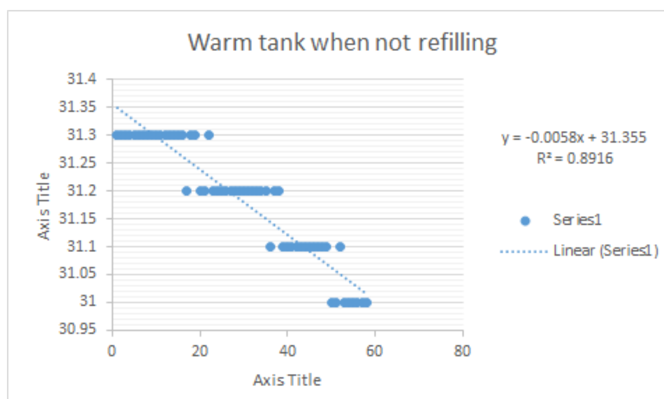

Figure 4 shows the temperature against time for the warm tank for F_r = 0

As and when the refilling occurs the fresh water is circulated through the warm tank first, hence there is a loss in the temperature of the warm tank. This can also be modelled by a linear equation. The warm temperature data range is selected when the value of F_r is 1. The linear regression is used to fit the curve to the raw data and hence, the temperature variation equation for heat loss in warm tank when F_r is 1 is obtained.

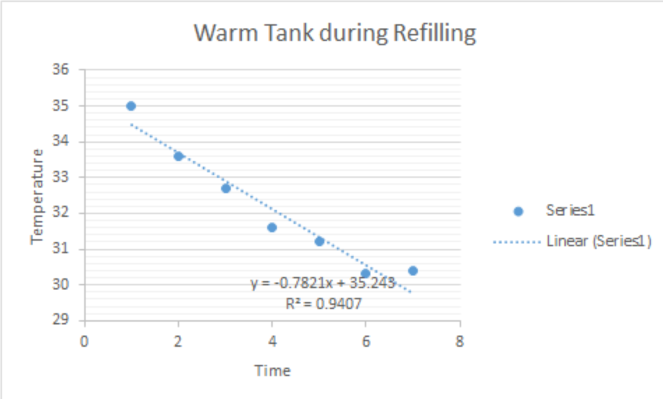

Figure 5 shows the temperature against time for $F_r = 1$

The cold tank gains the heat energy from the surroundings and hence the temperature of water in the tank increases with time. The value of conductivity of the tank can be obtained by fitting a linear line to the range of data where the F_c is 0 and no one uses the cold water. The slope of this line gives us the value of conductivity of the tank. The intercept of the line is the value of LC_T.

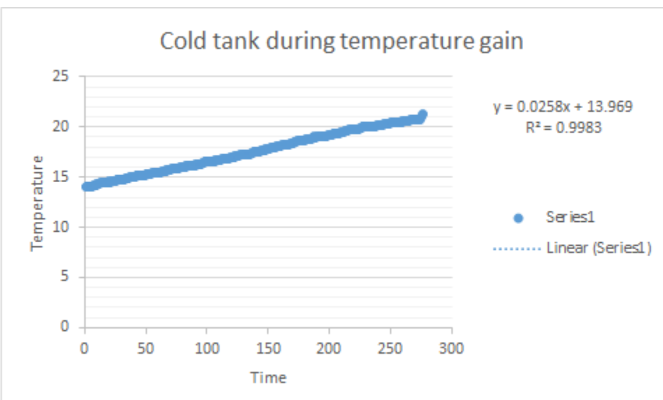

Figure 6 shows the temperature against time for cold tank

The cold tank equation during the cooling process can be obtained by selecting the range of cold tank temperature where the value of the F_c is 1 and then fitting a linear curve to it. The intercept of the curve is the value of UC_T. This has been illustrated in the figure 7.

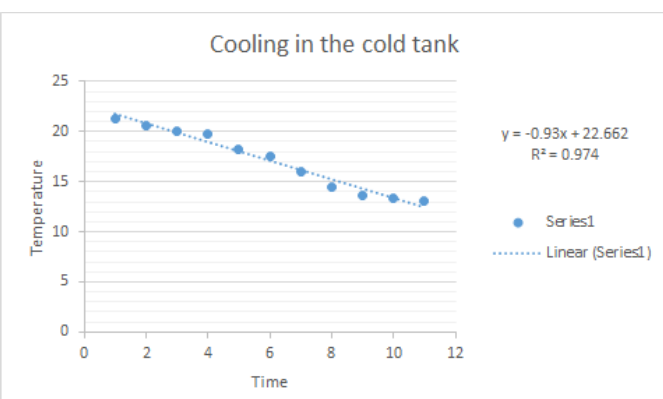

Figure 6 shows the temperature against time for cold tank

The refilling rate of the water dispenser can be obtained from the collected data by checking the fields of HighWater, MeanWater and LowWater flags. The rows where F_r is 1 are first selected and then the time is calculated for the value of the flags HighWater, MeanWater and LowWater flags changing from 0,1,1 to 1,1,1. This change is equivalent to adding 7 litres of water into the tank. Hence, the refilling rate can be obtained by

$$R_t \quad = \quad (7/TimeTaken) \quad L/mins. \tag{1}$$

The water outlet rate of the three tanks are fixed as 0.028L/s, 0.029L/s and 0.023L/s for the hot, warm and the cold tank respectively. These values combined with V_h, V_w and V_c gives us the value of the total time of the button press which controls the F_r flag of the simulator. The configuration parameter of the k_H, k_W and k_C are determined with the help of a python program as shown in the figure 1. The python program also helps to determine the values of the $T_H, u/T_H, l$ and $T_C, u/T_C, l$ for the starting of the dispenser. It also gives a csv file as output consisting of the volume of the water used in every minute and the tank from which water was used. Hence, it is assumed that the rest of the parameters remaining constant the only parameters required by the simulator and vary across each dispenser are as follows:

1) The upper and lower threshold temperature of the hot tank for regular heating
2) The upper and lower threshold temperature of the cold tank for regular cooling
3) The starting temperature of the warm tank
4) The value of conductivity for the hot tank, warm tank and the cold tank

The time of power saving of the dispenser is taken as an user input for a particular dispenser. The simulator program gives an output csv file consisting of the following parameters:

1) The temperature of the hot tank
2) The instantaneous volume of water in the hot tank
3) The temperature of the warm tank
4) The instantaneous volume of water in the warm tank
5) The temperature of the cold tank
6) The instantaneous volume of water in the cold tank
7) Total volume of water used by the simulator
8) Status of the flags of refilling, power saving, heating, cooling

II. Simulation Results

The parameters used in the simulations are $UH_T = 99.1$, $LH_T = 97.1$, $SW_T = 31.3$, $LC_T = 13$,$UC_T = 21$,$k_H = 0.047$, $k_W = 0.0058$ and $k_C = 0.0258$, $R_t = 0.58$, $r_t = 100$, $FW_T = 25$. The equation for heating is as follows:

$$T_h = 0.72 * time + 97.1 \tag{2}$$

The equation for the cooling is given by:

$$T_c = -0.93 * time + 21 \tag{3}$$

The equation of temperature drop in the warm tank when the water is refilling is given by:

$$T_w = -0.7821 * time + 35.2 \tag{4}$$

Based on the above parameters the simulation is compared with the raw data obtained from an actual dispenser. These are categorised into simulation under power saving mode, simulation for regular heating and cooling and the simulator under use with refilling.

- Power Saving Mode results:

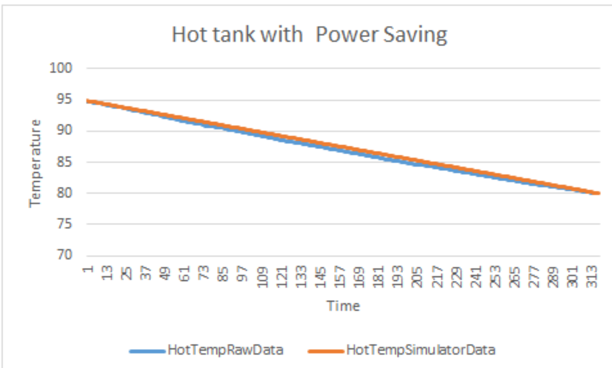

Figure 8 shows the results of hot tank in power saving mode

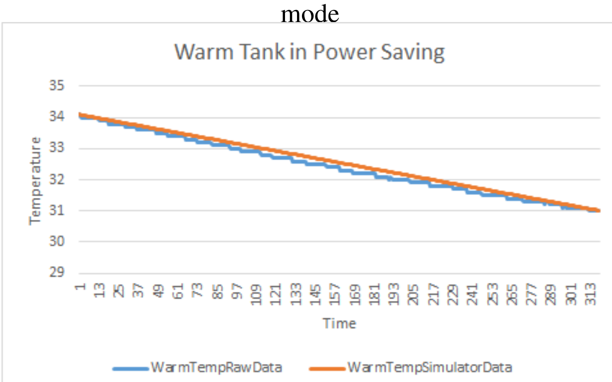

Figure 9 shows the results of warm tank in power saving mode

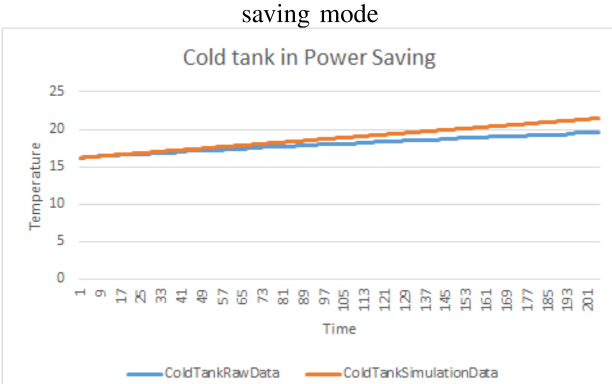

Figure 10 shows the results of cold tank in power saving mode

- Simulator for regular heating and cooling

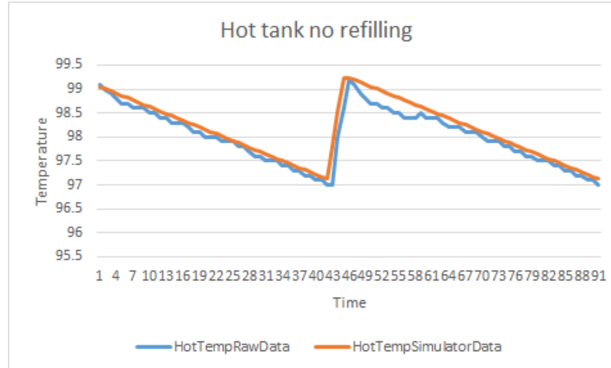

Figure 11 shows the results of hot tank in regular heating

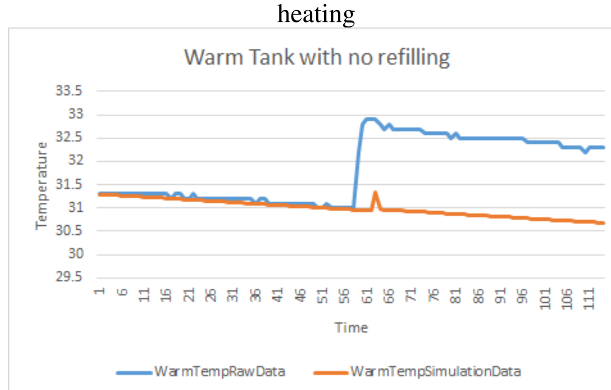

Figure 12 shows the results of the warm tank

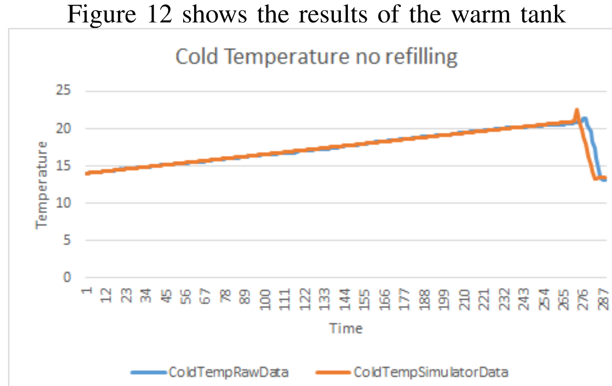

Figure 13 shows the cold tank results in regular cooling

- Simulator with refilling

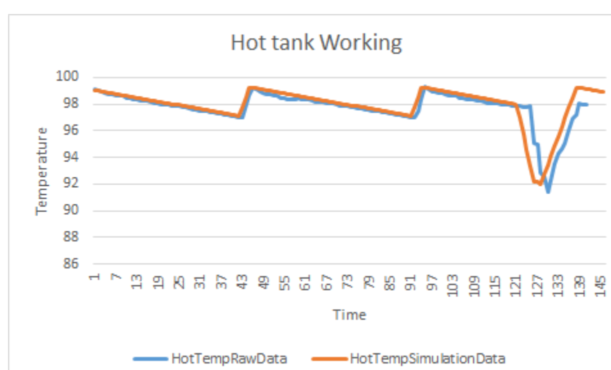

Figure 14 shows the results of the hot tank functioning

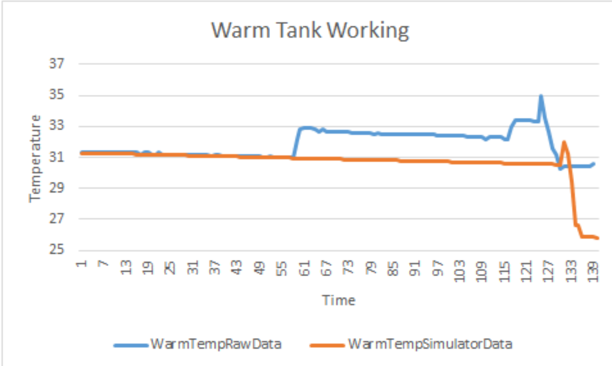

Figure 15 shows the results of the warm tank functioning

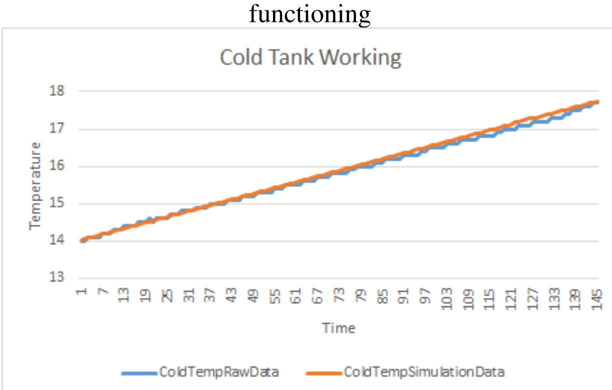

Figure 16 shows the results of the cold tank functioning

- Volume of Water used in simulator

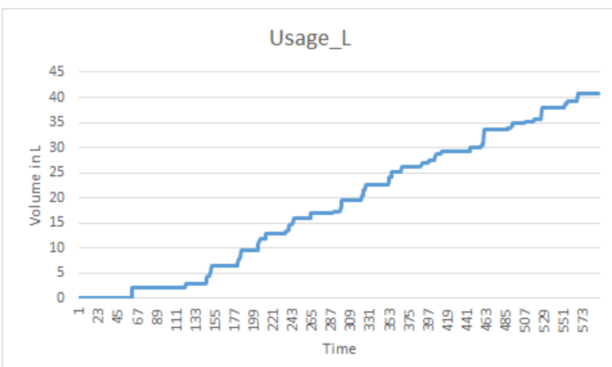

Figure 17 shows the volume of water used by the simulator in non power saving mode

- Temperature in the tanks of the simulator

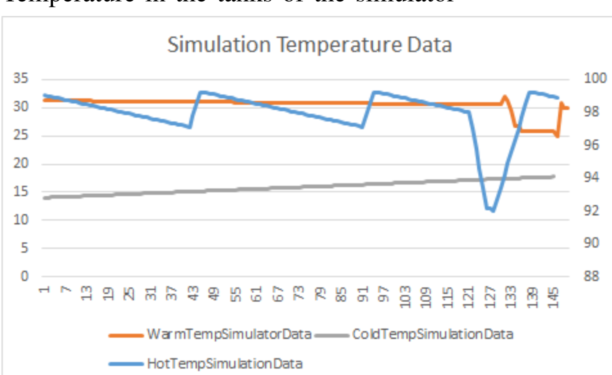

Figure 17 shows the temperature variation by the simulator in non power saving mode

III. Result Analysis

As it can be seen from the above presented figures for the comparison of the actual data collected from a dispenser in comparison to the data generated by the simulator. The shows how the temperature is varying in the power saving mode for the hot water tank. The raw data and the data from simulator almost have a perfect fit. Similarly, for **figure 9** the warm tank raw data as seen has almost a completely perfect fit with the simulator data. In case for the cold tank the slope in which the curve is generated by the simulator doesn't have a completely perfect fit, but since we can see in the **figure 13** the data seems to fit completely perfectly hence, the problem lies in the raw data collection and not the simulation function.

For the regular heating and cooling the simulator graphs show a perfect fit with the raw data as seen in the **figure 11** and **figure 13**. The warm tank although doesn't have a perfect fit with the graph for the raw data, since the sensor working in the collection of raw data seem to be faulty during that period, showing us a temperature rise in the tank although no one uses the water from the tank during that period. Hence, the simulator is unable to detect this sudden change in the data and shows the normal functioning of the data as in the ideal case.

For the **figure 14** showing the refilling of water to the hot tank, the raw data and the simulation data have a perfect fit but since the simulator has the $r_t = 100$ and the raw data has a varying time for the refilling period, the data for the simulator hence shows the refilling to be happening earlier as compared to the collected data. For the **figure 15**, the same problem for the sensor data collection is happening and although the slopes for the graphs are the same the temperatures as such are varying by a significant amount. For the figure 16, the cold tank is not affected by the cooling process and hence, is shown by the raw data and the simulation data.

Since, the collected data doesn't record the amount of water used in the tanks the **figure 17** doesn't show the volume consumption of collected data.

Acknowledgment

References

[1] W. Z. Cheng, R. G. Cheng, and S. Y. Chou, "Power-saving for IoT-enabled water dispenser system," *42nd International Conference on Telecommunications and Signal Processing (TSP)*, Budapest, Hungary, July 1-3, 2019.