A Feasibility study of PCR Control Using Cloud Computing

So-Yeon Lee¹, Jong-Dae Kim¹,², Hye-Jeong Song¹,², Yu-Seop Kim¹,² and Chan-Young Park¹,²*

¹Department of Ubiquitous Computing, Hallym University, Korea
²Bio-IT Research Center, Hallym University, Korea
{ye0721on, hjsong, yskim01, kimjd, cypark}@hallym.ac.kr

Abstract. Polymerase Chain Reaction (PCR) is a molecular biologic technology capable of diagnosing and analyzing diseases by replicating or amplifying specific regions of DNA. The PCR device we had previously developed and researched controlled the device using wired or wireless communications configured with a local-host system. However, the above system is subject to user device and experimental data accessibility restrictions. Therefore, in order to solve the above problems, the present study investigated the possibility of controlling PCR apparatus using cloud computing environment, developed a PCR device monitoring program that allows the user to verify the experimental data of current or past PCR devices. As a result of the research, there was no problem in starting and ending the PCR device. However, when the cloud computing environment interferes with the temperature control of the device, the temperature control problem can be obtained.

Keywords: Cloud Computing, Round-trip

1 Introduction

Polymerase Chain Reaction (PCR) is a molecular biologic technology capable of diagnosing and analyzing diseases by replicating or amplifying specific regions of DNA [1].

Even recent simple electronic devices require a graphical user interface (GUI) that utilizes graphics capabilities and access to related data over the Internet. In this study, we have developed a method to control PCR through the cloud to improve user convenience and reduce system development cost [3]. This method allows multiple PCR thermocycler to be driven in a single cloud by recognizing multiple PCR thermocycler as an independent device through the cloud. In addition to controlling the PCR device through cloud computing, it is possible to manage the data related to the PCR device, such as the chamber temperature and status of the device. However,
if the communication speed between the PCR device and the cloud is slowed or delayed, the operation of the PCR device may be disrupted.  

2 Materials and Method

A total of three emulators were used in this study. The devices connect to the cloud database individually over the network. We also developed a monitoring program that sends the data of the devices to the cloud or sends the received data to the equipment to control the temperature and view the temperature in real time. This program not only shows the progress of PCR protocol in real time, but also can check past experiment history easily through graph.

**Fig. 1. Firebase and Arduino work flow**

Figure 1 shows a workflow diagram of the PCR emulator device and the cloud. A device that is connected to the network used in the research is connected to the cloud to exchange data. When the first program runs, the device connects to the cloud and waits for a work order from the cloud. When you check the 'Run command' from the cloud, it starts to send data to the cloud as soon as you issue a command to the device. At this time, the data to be temporarily stored is stored in a queue form. When the device is in operation, the device's data is stored in a queue and is considered to be operational if there is no more data to send to the queue. At the same time, the data stored in the queue is sent to the cloud. If the data in the Queue is empty and no more data is accumulated, it stops transmitting data to the cloud. If the PCR emulator finishes working and no more data is sent to the cloud, all work has been done, so wait until there is a device operation command.

When controlling PCR externally, the communication speed between the client and the cloud environment becomes very important. When the PCR protocol is transmitted to the PCR device at a time, there is no big problem even if the
communication environment between the client and the cloud is poor. However, in case of the PCR which transmits the protocol step by step, the communication speed is slow or delayed, Can be. For this reason, experiments were conducted to measure the time taken to exchange data between the device and the cloud database.

3 Result

Figure 2 above shows a PCR monitoring program. The program has been developed for users to visualize real-time or historical data and for data statistics. If you want to see past experimental results, select 'Ended Thermograph'. When this item is selected, the items of 'Date' and 'Number' are activated, and Number means the number of times of experiment on that day. Select the experiment date and number of the device and press the 'Apply' button. The sensor values of the date are displayed in the graph. When you select a protocol in the cloud and issue a command to run a PCR, you can see the status of the currently running protocol in this program. If you select 'RealTime Thermograph' and press 'Apply' button, you can see the temperature of chamber according to current protocol in real time, and the result is stored in 'Number' 0th. After the selected protocol has been completed, it will be executed again. 'Boxplot' means the number of executions per connected device. In Boxplot, the center line represents the median of the data, the upper and lower edges of the box represent the upper quartile (Q3) and the lower quartile (Q1) of the quadrants, and the amount of lines connected to the box. The end means the minimum value and the maximum value of the data excluding the singular point.
Table 1. Data Rate by data length

<table>
<thead>
<tr>
<th></th>
<th>Data1</th>
<th>Data64</th>
<th>Data128</th>
<th>Data256</th>
<th>Data1024</th>
<th>Data2048</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIN</td>
<td>0.386064</td>
<td>0.385231</td>
<td>0.387225</td>
<td>0.389334</td>
<td>0.400049</td>
<td>0.41081</td>
</tr>
<tr>
<td>MAX</td>
<td>12037414</td>
<td>46.32331</td>
<td>23.48343</td>
<td>24.19815</td>
<td>22.32283</td>
<td>6.735785</td>
</tr>
<tr>
<td>AVERAGE</td>
<td>0.457295</td>
<td>0.458813</td>
<td>0.462393</td>
<td>0.464804</td>
<td>0.486865</td>
<td>0.527903</td>
</tr>
</tbody>
</table>

Table 1 shows the maximum, minimum, and average time of data rate for each data length. The minimum value was measured between 0.3 and 0.4 seconds, and the peak value is at least 6 seconds, which is relatively longer than the average of about 0.5 seconds. Generally, the PCR chip operates at about 10 degrees per second and controls the temperature within 0.5 degree, so the average speed of sending and receiving data should be 40ms. However, the experimental results show that the average speed of sending data to the firebase using the sequence number and fetching the data stored in the firebase has been measured to be slower overall, from 0.4 to 0.5 seconds. Although the average speed was measured slower than the required speed, the device used in the study was not operated in real time, so there was no problem in running the device. If there is a delay in the communication speed, there may be a delay in executing the next protocol, but there is no problem in the device during the delay of the protocol.

4 Conclusion

In this paper, we developed a monitoring program to monitor the progress of the PCR device and experimented with the possibility of controlling the PCR device in the cloud. As described above, the communication speed between the cloud and the device was measured slower than the required speed. In the case of controlling a device using Cloud, a major problem was whether the delay in communication speed caused a problem in device execution. Applying real-time and deadline requirements, the processing of the protocol is in seconds, so it does not matter in the cloud. However, in the case of temperature control, a problem may arise because a processing time of about 40ms is required as described above.

The proposed system is more efficient than the existing PCR system composed of local host system because protocol and result data are managed in the cloud. In the case of a PCR device operating with real time, it is not recommended because the communication speed may cause problems in the temperature control part.

References