Video and USB Transmission Devices for Cloud Desktop Service

Chanho Park(✉) and Hagyoung Kim

Cloud Computing Research Department,
Electronics and Telecommunications Research Institute (ETRI),
218 Gajeong-ro, Yuseong-gu, Daejeon, Korea
{jangddaeng,h0kim}@etri.re.kr

Abstract. This paper shows a video and USB transmission device for high-quality cloud desktop service. In a general software-based desktop service, an application such as a game cannot be serviced normally due to a long latency. To solve this problem, a passthrough virtualization system which assigns a graphic card to each user is used and hardware devices which transfer video and USB signals to the network are developed. In the hardware device, a simplified switch is adopted to support multi-user. A cloud desktop service using these devices can provide a satisfactory service in an application where the latency becomes a problem.

Keywords: Video · USB · Latency · Simplified switch

1 Introduction

Cloud Services can be divided largely into SaaS, PaaS and IaaS. The service ordinary users often encounter is the SaaS. DaaS (Desktop as a Service) is similar to SaaS and this service provide a personal desktop virtually to a user. These services (SaaS and DaaS) can support many users on a single server, but has limitations on the application which they can provide. For a high-resolution multimedia service, as concurrent users increase, it becomes more difficult to provide the service. Indeed, for an application such as a game which requires low response time, it is almost impossible to provide the service. This is mostly because of the long delay time of graphic processing.

There are several methods for graphic virtualization in a server. Typical examples are full virtualization, mediated passthrough [1], and direct passthrough [2, 3]. Additionally, hybrid styles such as gVirt are also possible [4]. Among them, direct passthrough method shows good performance close to the non-virtualized system, even though it can support fewer users because one GPU has to be assigned to every VM. And it has advantage that chipper video cards can be used.

Passthrough feature is not only applicable to GPUs but also to other peripheral devices. DaaS user can directly interface to the VM without emulation if the passthrough feature is applied to USB devices. A system called "green PC solution" which applies passthrough feature to graphic cards and USB ports is developed. In this system, a graphic card and a USB port are assigned to each VM and output of graphic card and

Y. Zhang et al. (Eds.): CloudComp 2015, LNICST 167, pp. 184–188, 2016.
DOI: 10.1007/978-3-319-38904-2_19

USB port is connected to the user with a direct cable. In this paper, to substitute the direct cable with an IP network, an add-in card for a server and a client device for a user are proposed. Additionally, a switch structure in the add-in card is described.

2 Video and USB Transmission Devices

In a common cloud service, software based solution is installed in a server and a service user uses a device called “Thin client” or “Zero client”. The proposed system is equipped with multiple graphic cards and a server card which transmit the graphic card output port and USB port signals to a network. For a service user, a dedicated “Zero client” is used for short response time, even though a common “Thin client” can work. Figure 1 shows the difference between the common cloud and the proposed cloud service systems. The server card can be implemented in several ways.

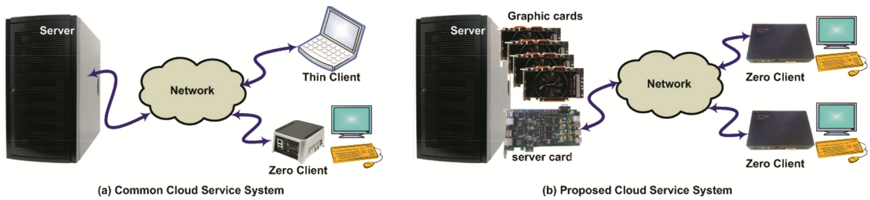

Fig. 1. Comparison of the common cloud system and the proposed system

Figure 2 shows an example of a server card using commercial chips which convert video and USB signals into network packets. This example is a server card which supports 4 users. Each “video & USB processor” receives video and USB signal from the port and convert them into network packets. It also receives network packets which contains USB information, convert the packets into USB signals, and sent the signal to the USB ports. Because this card has only one network port, each “video & USB processor” has different IP address and the network packets are routed by the switch chip. Video transmission chip and USB over IP chip can be separately implemented. The client device is implemented with the same chip of the server card without a switch because it is only for one user. FPGA can be used instead of commercial chips [5]. In this case, switch can be implemented in the FPGA together. In some case, a commercial chip has the function of a graphic card. Then, graphic cards are not required and the chips communicate with the server using PCI-express interfaces. However, these chips cannot be used for an application requiring GPU with high performance, so these chips are used as BMC in the server management system or as a processor for KVM over IP system.

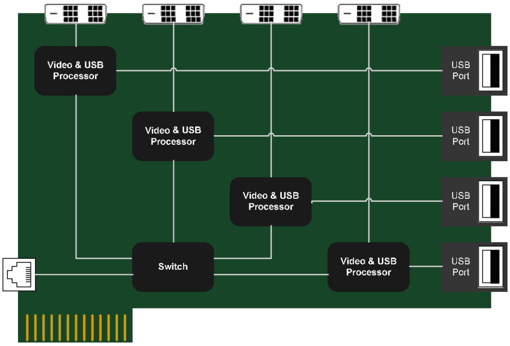

Fig. 2. An example of a server card

3 Simplified Switch Structure

Since the server card has to support multiple users, it must include a switch. Generally, a common switch chipset is used. However, switch chip becomes expensive as the number of port increases. And, in this case, it is not necessary for the chip to act as a general switch. Because video and USB signals of VM are only required by a user in an external network, communications between inner ports are not necessary. In the switch, all packets from inner ports should go to the external port and packets from the external port should be distributed to the inner ports. Therefore, the simplified form of switch can be used instead of a general switch chip.

Figure 3 shows a simplified model of a switch chip. All packets from inner ports are delivered to external port without any processing (or with simple congestion control algorithm). All packets from external port are distributed to inner ports according to the destination IP address. For this, the IP information of "video & USB processors" connected to the inner ports must be recorded in the registers. If a received packet does not have a unicast IP as a destination, this packet is broadcasted to every inner port.

To simplify the structure of switch further, all packets from external port can be broadcasted to every inner port. It is shown in Fig. 3(b). Whether packets are received or dropped is the responsibility of the chips which are connected to the inner ports. By this approach, the switch function can be implemented very easily, but has some disadvantage of power consumption due to unnecessary packet delivery.

As mentioned before, if the server card is implemented with FPGA instead of commercial chips, switch function can be implemented in FPGA together. Because there are no separated chips, only one MAC is enough to accomplish the switch function. Not only the MAC address, IP address can be shared among VMs (the IP to send and receive video and USB signals, not the IP of VM itself). In this case, received packets from external port can be distributed according to the port number.

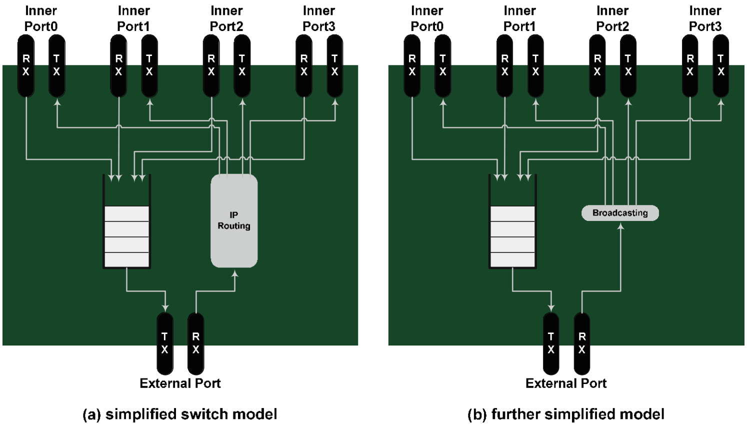

Fig. 3. Simplified switch model

4 Implemented Results

Figure 4 shows server cards supporting two users and client devices, each of which consists of a commercial chips and FPGAs. With these devices, the delay time of DaaS service on a LAN is tested. Each represent the delay time of 40 ms and 70 ms. The difference between the two delay times is mostly due to hardware codec implementation. In both cases, there is no problem to service an application which requires fast response time such as a game.

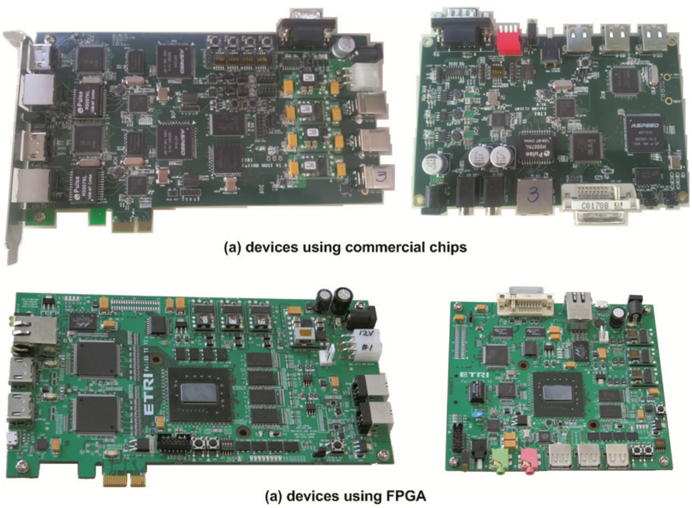

Fig. 4. Implemented server cards and client devices

5 Conclusion

This paper shows a video and USB transmission device for high-quality cloud desktop service. A passthrough virtualization system which assigns a graphic card to each user is used and hardware devices which transfer video and USB signals to the network are developed. In the hardware device, a simplified switch is adopted to support multi-user and to reduce design complexity and cost. A cloud desktop service using these devices has delay time less than 70 ms on the LAN and provides a satisfactory service in an application where the latency becomes a problem.

Acknowledgement. This work was supported by the ICT R&D program of MSIP/IITP. [B0101-15-0548, Low-power and High-density Micro Server System Development for Cloud Infrastructure].

References

1. Xia, L., Lange, J., Dinda, P., Bae, C.: Investigating virtual passthrough I/O on commodity devices. In: Proceedings of ACM SIGOPS, pp. 83–94 (2012)
2. Abramson, D., Jackson, J., Muthrasanallur, S., Neiger, G., Regnier, G., Sankaran, R., Schoinas, I., Uhlig, R., Bembu, B., Wiegert, J.: Intel virtualization technology for directed I/O. Intel Technol. J. **10** (2006)
3. Dong, Y., Dai, J., Huang, Z., Guan, H., Tian, K., Jiang, Y.: Toward high-quality I/O virtualization. In: SYSTOR (2009)
4. Tian, K., Dong, Y., Cowperthwaite, D.: A full GPU virtualization solution with mediated passthrough. In: Proceedings of USENIX Annual Technical Conference, pp. 121–132 (2014)
5. Park, C., Kim, H.: Low latency video transmission device. In: Proceedings of International Conference on Information Science and Application, pp. 217–222 (2015), International Conference (2014)