Remoting Peripherals using Memory-Mapped Networks

S. J. Hodges*, S. L. Pope*, D. E. Roberts*, G. E. Mapp* & A. Hopper*†
*Olivetti and Oracle Research Laboratory
†Cambridge University Engineering Department
Trumpington Street,
Cambridge,
England.
{sjhodges, spope, droberts, gmapp, ahopper}@orl.co.uk

Abstract

Memory-mapped networks such as Scalable Coherent Interconnect (SCI) [5] and Memory Channel [4] offer a new method for constructing network peripherals by remoting a host's IO bus. This paper details our experiences from building such an endpoint, and examines how greater support may be provided.

1 Introduction

At ORL we have experimented with an *exploded workstation* configuration, where low-cost multimedia peripherals known as *network end-points* are attached directly to an ATM network. End-points include video [3], audio, compact liquid-crystal display [1] and storage devices [2]. Whilst our experience of directly networked peripherals has been encouraging, there were some shortcomings:

- The backend of a networked peripheral must be heavily re-engineered to interface with the network. This development effort means that networked peripherals are usually behind the state of the art.
- The use of networked peripherals requires driver reimplementation, even if the peripheral is supported by the system when directly attached.

A network interface which presents a memory-mapped abstraction and passes arbitrary IO bus cycles over a network should allow a networked peripheral to be directly controlled from a host workstation as if the peripheral were on the host's IO bus. In effect, this could be described as the **remoting of a host's IO bus over the network.** We chose to investigate this possibility using SCI technology from Dolphin Interconnect Solutions [6]. This hardware is PCI based and can therefore be used in a number of workstations including standard PCs.

2 Remoting the StrongTile

The StrongTile is an in-house PCI display device designed to replace the existing VideoTile [1]. Our aim was to remote this card by placing it in a dumb PCI backplane with an SCI card, and to control it directly from a host workstation across an SCI network, as illustrated in Figure 1.

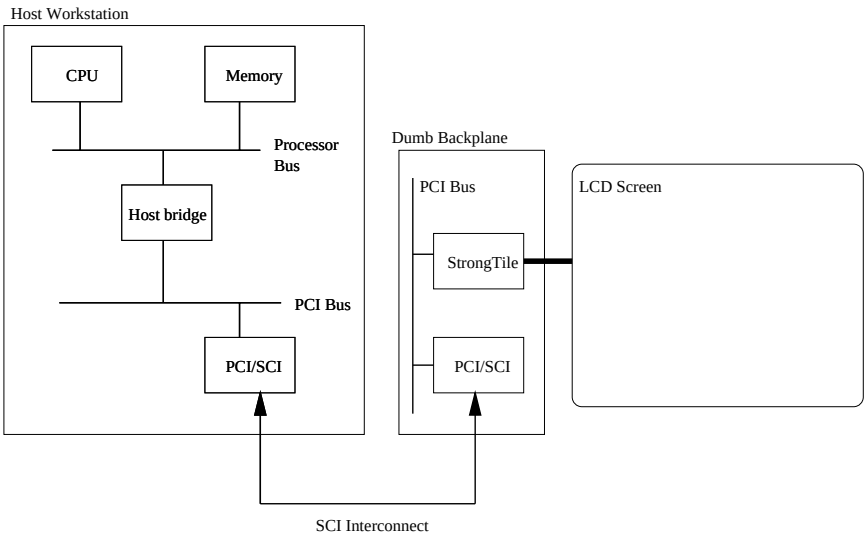

Figure 1: Remoted StrongTile configuration.

In our hardware configuration, the SCI card in the host workstation appears to the host PCI bridge, BIOS and device drivers as a network card rather than a graphics card and hence it was not possible to configure the StrongTile device from the host workstation.

Our solution was to place a processor card on the remote PCI bus to perform configuration for the remote SCI card and StrongTile device. The StrongTile device was then mapped into the local host's address space and software running on the local host was able to drive the remote StrongTile as if it were a local device. The performance of the remoted StrongTile is comparable to when it is located in the host PCI bus. A picture of the StrongTile displaying an X-windows session is given in Figure 2.

3 Conclusions

Our experiences of interfacing directly with hardware have outlined some outstanding issues which remain to be resolved. In particular, our current technique removes the requirement for effort at the peripheral end, but is not transparent to the host and requires driver modifications in order to successfully operate.

To be transparent, the network card must be capable of emulating the remoted card itself or else appearing as a bridge to a bus containing the remoted card. Work has begun at ORL on building such a device.

References

[1] M. D. Addlesee. The VideoTile, 1996. http://www.orl.co.uk/tile.html.

[2] A Chaney, I Wilson, and A Hopper. The design and implementation of a raid-3 multimedia file server. In *NOSSDAV'95*, April 1995.

[3] D. J. Clarke. The Medusa video brick: an ATM camera. Technical Report 4, Olivetti & Oracle Research Laboratory, 1995.

Figure 2: The StrongTile.

[4] M. Fillo and R. Gillett. Architecture and Implementation of Memory Channel 2. *Digital Technical Journal*, 9(1), 1997.

[5] IEEE. Standard for Scalable Coherent Interface, March 1992. IEEE Std 1596-1992.

[6] S. Ryan, S. Gjessing, and M. Liaaen. Cluster communication using a PCI to SCI interface. In *IASTED Eighth International Conference on Parallel and Distributed Computing and Systems*, October 1996.