M.I.T. Laboratory for Computer Science

Request for Comments No. 215
December 4, 1981

A Project to Explore the Networking of Low-Cost Personal Computers

by Wayne C. Gramlich

1. Introduction

Shortly, the Computer Systems Research group (alias Computer Systems Structure Group and Computer Systems and Communications Group) will be obtaining three IBM 5150 personal computers. Each IBM 5150 consists of an Intel 8088 16-bit processor, up to 192 kilobytes of primary memory, up to 2 floppy disk drives, a CRT screen, a keyboard, *and* an RS-232 serial interface. The cost of an IBM 5150 with full memory and both floppy disk drives is approximately $5000. The question arises of how to attach these low-cost personal computers to the MIT network environment. In the past, a computer has been attached to the network via a hardware network interface that plugs directly into both the computer and the network. The typical cost of such a hardware network interface is approximately $3000. Alternatives to a hardware network interface need to be explored, since the cost of a hardware network interface is almost as much as the personal computer. These alternatives will undoubtably reduce hardware costs at the expense of increased software complexity. In addition, it must be realized that there is an upper bound on the software complexity due to the memory and processor limitations of the personal computer. The goal of this project is to examine the hardware/software trade-offs involved in attaching personal computers to the MIT network environment.

2. The Basic Approaches

Each personal computer can not be cost effectively connected directly to the MIT network environment. However, it may be acceptable to share one network interface between several personal computers. This divides the cost of the network interface between the personal computers. A *Personal Computer Network Interface* (PCNI) is a device whose sole purpose is to connect one or more personal computers to a local network.

An alternative to the PCNI is to connect the personal computers to a piece of hardware that is already uses a local network interface for some other purpose. In this approach, the cost of the local network interface is divided between the personal computers and the other function that the hardware is performing. There are currently two pieces of hardware that are likely candidates for this approach. The Terminal Interface Unit (TIU) is used to connect terminals to the local network for remote login to local network hosts. The gateway (GW) is used to interconnect two or more local networks. Each of these devices consists of an LSI-11 with at least one V2 LNI.

Initially, it is proposed that the personal computers be connected to the GW/TIU via 9.6K baud RS-232 asynchronous serial I/O lines. A serial I/O port for an LSI-11 costs approximately $100/line.

There are two primary tasks that are desirable to do with the personal computers. First, it is desirable to transfer files between the personal computer and the local network. Second, it is desirable to be ab!^e to remotely log in to local network hosts. The other secondary tasks are usually built in terms of file transfer an remote login. For example, a mail system needs the use of a file transfer package.

An initial examination of the problem shows at least three basic approaches. First, the gateway may be modified to treat the personal computers as a subnet. Second, the PCNI may be developed as a front-end to the local network. Third, software may be developed for the personal computer that uses the TIU without any modifications to the TIU.

2.1. The Gateway Approach

The approach of modifying the gateway requires that all of the protocols necessary to interact with the local network be implemented on the personal computer. The required protocols are 1) Internet Protocol (IP), 2) User Datagram Protocol (UDP), 3) Trivial File Transfer Protocol (TFTP), 4) Transmission Control Protocol (TCP), and 5) remote login protocol (TELNET).

The gateway approach is further sub-divided into the following components.

~ The protocol to be used over the serial line between the gateway and the personal computer must be specified. About the only problem with this protocol is that it must be able to re-synchronize in the face of character lossage errors. The checksum contained each internet packet should suffice to detect transmission errors between the gateway and the personal computer.

~ The gateway must be modified to use the serial line protocol. The set of serial lines connected to the personal comuters is expected to be treated as one logical sub-network to the gateway. One case that must be considered is how to send internet packets between two personal computers attached to the same gateway.

~ The serial line protocol must be integrated into the personal computer environment. A partial list of relevant issues that must be dealt with are:

  * Is the serial interface dealt with directly by the software or through the operating system?

  * Is the serial interface interrupt driven or polled?

  * Will it be possible to transmit/receive on the serial line while accessing the disk?

  * Will it be possible to transmit and receive simultaneously?

~ A remote login program must be developed. This program will use the TCP and TELNET protocols. It is expected that the TCP implementation will be tailored to the TELNET application.

~ A file transfer program must be developed. This program will use the UDP and TFTP protocols although they may be integrated into a single package.

~ A mail sending/receiving program must be developed. The reception of mail must deal with the fact that the personal computers may be powered down when a piece of mail arrives.

~ Code to access a name server must be developed.

~ Code to print out a file should be developed.

2.2. The Front-End Approach

The front-end approach requires that all or most protocols needed to interact with the local network be implemented in the front-end processor (PCNI). The front-end processor interacts with the personal computers via a protocol specialized to maximize bandwidth utilization between the front-end processor and the personal computer.

It has been decided not to pursue the front-end approach unless the gateway approach appears not to work. Briefly, the reasons for this decision are:

- The front-end approach is not as cost-effective as the gateway approach since the front-end approach requires a dedicated processor. This is in contrast to the gateway approach where the gateway is also used to interconnect subnets.

- The front-end approach seems to be justifiable only if the gateway approach has serious deficiencies.

2.3. The TIU Approach

The TIU approach is to the utilizes the TIU without modification. Remote login is accomplished simply by running a program on the personal computer that routes keyboard input directly to the TIU and output from the TIU directly to the personal computer display. (Frequently, such software is commercially available for the personal computer.) File transfer is accomplished between the personal computer and a remote host by using an as yet undeveloped protocol. It is necessary for the remote host to implement one half of this protocol and the personal computer to implement the other half of the protocol.

This approach will be useful as a stepping-stone for initially transfering information to the personal computers. However, this approach does not meet the goal of making the personal computer a full participant in the inter-network protocol family.

3. Hardware Projects

There are three basic hardware projects that can be undertaken.

- Hardware can be developed to try to improve the performance of the RS-232 line from 9.6K baud to a higher baud rate.

- Local network interfaces can be developed for the personal computers. David Feldmeier is currently debugging an LNI interface for the S-100 bus.

- A low-cost micro-network can be developed strictly to interconnect personal computers. There is currently a proposed design for a tree-shaped ethernet that consists of approximately 10 IC's per cluster of 8 personal computers.

The intial plan is not to undertake any hardware development. The plan will change if 1) it becomes necessary to have a particular piece of hardware, or 2) someone comes along who is really interested in doing one particular hardware project.

4. Programming Environment

All of the software projects require that some software be developed on the personal computer. Hence, it is going to be necessary to provide a software development environment for the personal computers. It is proposed that most, if not all, software be written in a high level language. For the IBM 5150, two high-level languages manifest themselves. The IBM 5150 comes with a resident Pascal compiler. There exists a C compiler that generates code for the 8088 that runs on a PDP-11 under the Unix operating system.

Currently, there exist versions of all the protocols written in C. However, it is unlikely that any of the protocols can be transported to the personal computer without a complete rewrite. Thus, there is no compelling reason to use C in preference to Pascal on the IBM 5150. It may be the case that the Pascal compiler does not produce efficient enough code to implement the protocols. One of the first tasks that must be accomplished is to benchmark the Pascal compiler for the IBM 5150.

If it is decided to use the C compiler the following tasks must be performed:

1. A loader must be developed for the IBM 5150 that permits code cross-compiled on the PDP-11 to run on the IBM 5150.

2. A set of library routines to interface to the IBM 5150 operating system must be developed. It may be desirable to use the CP/M-86 operating system instead of the IBM operating system.

3. (Optional) The C compiler may be ported from the PDP-11 running Unix to the IBM 5150.

The programming environment for the IBM 5150 is very critical to the entire research project, since it lies in the critical path of all solutions. Thus, it will be necessary to allocate enough manpower to ensure that the programming environment is in fact developed.