Adelin M. Clesse

Hewlett Packard 3352 Gas Chromatograph System to HP 3000 Link

(Micro Processor Communication Link for Data Collection)

A communication link has been established from an HP 3352C Gas Chromatograph Data System and an HP 3000CX. This link allows scientists and technicians running Gas Chromatographs to automatically transfer their data to a 3000 Disc file.

The purpose of this paper is to describe the hardware/software problems which were encountered and our solutions to those problems. Of particular interest is the fact that a "link controller" based on a microprocessor was designed and constructed to interface the two systems.

BACKGROUND

In December, 1975, the Scientific Systems Development section of Norwich Pharmacal Company launched a Laboratory Automation project which was to aid the Company in its Drug Research and Development activities. This commitment to computers in the laboratory came as a result of an extensive survey and planning period which closely involved all persons who were potential users of the system.

The overall configuration which was selected to satisfy the diverse needs is shown in Fig. 1 and is composed of three separate Hewlett Packard systems.

The first system, designated as the host, is the HP 3000. The configuration for this system includes: 48K memory, two 7905 discs drives, card reader, line printer and the R.J.E. subsystem. The function of this system is to provide time sharing service to users in the Research Center running a mix of BASIC, FORTRAN, EDITOR, assorted data entry programs, limited Image/Query applications and remote job entry to an IBM 370/145.

The second system is an HP 9600 coupled to the 3000 using the Programmable Controller subsystem. This configuration includes: 21MX with 32K of memory, 32 channel Analog/Digital converter, paper tape reader, and an assortment of digital interface cards. The purpose of this system is to automate 12 analytical instruments (Auto-analyzers, Coulter counters, spectrophotometers, electronic balance, Differential Scanning Calorimeter and CHN Elemental

Analyzer). Programs to support these various instruments are presently being written by SSD personnel.

The third system is the HP 3352 Gas Chromatograph Data System. This "turnkey" system consists of a 21MX with 32K of memory, paper tape reader, five terminals for reports and twelve Analog/Digital modules required to attach as many instruments. The purpose of this system was to automate the time consuming task of data analysis for a specific group of analytical instruments. This group includes gas chromatographs, liquid chromatographs and spectrodensitometers. The software as provided by HP collects data from each instrument and analyses that data (i.e. integrates peaks, performs baseline corrections, and calculates concentrations). The 3352 system then outputs a final report which identifies the peaks and indicates their retention time, area and concentration.

The vendor software does not provide a convenient way to manipulate data beyond the final report phase. This is in part due to the diverse needs of the many users. Since the 3000 could provide this capability, it was decided to perform these additional calculations using that system.

This paper deals primarily with the link which was established from the 3352 to the 3000 to make these additional manipulations possible.

GROUND RULES OR RESTRAINTS

Due to the problems encountered when one attempts to modify existing vendor software, it was decided that we would avoid making any changes to the present system software. On the GC Data System, we would have to use a single user BASIC interpreter to handle the transfer of data out of that system. For the 3000, we would use present existing system software and FORTRAN to handle data entering the system through the asynchronous terminal multiplexer.

Since the 3352 BASIC can process data for only one GC run at a time (other requests are queued up), the transfer of data would have to be as fast as possible to minimize any delay in the start of a new instrument run. One other problem which we encountered was that the single user BASIC would not allow us to perform Input/Output to more than one serial device. It did, however, permit the use of parallel interfaces defined as paper tape reader and punch.

Since the link would be terminated using one of the multiplexer ports on the 3000, it would be necessary that we programatically define the link as a "data device". Once the data had been entered and stored on the 3000, it would have to be conveniently available to all users immediately after the termination of that particular run.

Since the file space on the 3000 is limited, it was decided that the data would remain available for a limited period of time (7 days) after which it would be automatically purged.

The last requirement was that, if a problem develops or maintenance be required on either system, the other should continue to operate albeit without transfer capabilities. For the 3352 this meant that if the 3000 was not available gas chromatographic data analysis could continue to be performed without automatic storage of the generated data.

SYSTEM DESCRIPTION

Figure 2 shows a block diagram of the link and will be helpful in describing data flow during transfer operations. The link controller shown in the diagram performs a number of functions:

1) Converts data from parallel to serial and serial to parallel format.
2) Buffers an entire record between the two computers.
3) Monitors current status of the 3000.
4) Maintains synchronization between the two systems.

This controller has been implemented with state of the art technology. The heart of the controller is a microprocessor with capabilities similar to many small computers. Amazingly, the cost of this device itself is only about 20 dollars. Other components in the controller increase the total cost somewhat. The microprocessor approach is still cost effective due to the flexibility acquired through its use.

The need for the controller may be more obvious to those who have tried to enter data through a multiplexer port from a peripheral device. Before sending data to a 3000, the device must ensure that the 3000 is ready to receive data. This requires at a minimum that the device file be opened and that a "READ" operations be pending at that particular moment. The time required to perform these operations is influenced by system loading. When speed is essential, unnecessarily long delays are undesirable. One feature of the multiplexer which we were able to make use of is the automatic transmission of a "DC1" when the 3000 is ready to accept data. This particular sequence is used to trigger 2640 type terminals. We have used it to indicate a 3000 ready-to-receive status.

In an idle state, the controller maintains synchronization with the 3000 by satisfying the pending read every fifteen seconds. The controller then waits for a new "DC1" from the 3000. This idle polling sequence allows the Fortran/3000 program to periodically check a flag file (every sixty seconds). If it finds this flag "set" the program will close opened files, release the multiplexer port and terminate itself. This program normally runs as a batch job with CS priority 23 hours/day.

When a gas chromatograph run is complete, the 3352 system calculates final results and displays these at a terminal near the particular instrument. After the final report has been printed, the single user BASIC is automatically invoked. A short BASIC program then requests the final report data from the 3352 system and passes that data to the controller a line at a time.

As each character is sent to the 3000 by the controller, an echo is received. This echo is then checked with the character sent. This verification process guarantees that the data stored on the 3000, located six hundred feet away, is identical to the data printed in the final reports.

If the HP 3000 does not respond to an attempted data transfer within fifteen seconds, the controller will time-out and indicate the failure to the 3352. If synchronization is lost with the 3000, any attempt to transfer data will be immediately aborted. For either condition the 3352 prints a message at the local terminal to indicate that the instrument data had not been stored.

When a complete report has been received by the 3000, the data file is locked and the updating is performed. After unlocking the file, a file number and date stamp are sent to the controller to be used by the scientist for subsequent access to his data. This information is conveniently displayed at the bottom of his final report.

A program is executed daily which discards all data stored for more than seven days. If extended storage is required, the data can be extracted from the GC link account and stored in the users own files. This technique allows us to keep the file down to a reasonable size (132,000 bytes).

To extract data from the file all that a user need do is write a short BASIC/3000 program which calls a FORTRAN subroutine. This FORTRAN subroutine opens the file and locks it to assure exclusive access, reads the required information and then passes that data to the calling BASIC program. Since the subroutine has been stored in the System SL, the scientist is only concerned with his run parameters and remains completely isolated from the required file manipulations.

TECHNICAL DETAILS OF THE LINK CONTROLLER

The use of a microprocessor for the link controller was not a part of the original design. The original design was based on a parallel to serial integrated circuit (UART) with a few other logic modules for control. It was only after a few unexpected complications came to our attention that we realized that our digital logic circuit was getting very complex and difficult to modify. Although the functions to be implemented by the microprocessor were complicated, our experience in solving computer problems made the solution to our interfacing problems more manageable than if we had used the very complex digital logic.

The use of the microprocessor allowed us to simplify the operations in both minicomputers. The link controller could now easily buffer records, check the echo on serial transmission and indicate to the 3352 if the 3000 was not available. The link controller is completely automatic in that no operator actions are required for daily startup. The link controller operates continuously.

The microprocessor which we chose for this application is the National SC/MP (Simple to use, Cost effective microprocessor). We chose this particular microprocessor for several reasons:

- Cost: An evaluation kit which included RAM, ROM, clock components and interface was available for 99 dollars.
- Software Development: A cross assembler was available on the G.E. Timesharing Network.
- Availability: The above evaluation kit was available from a local distributor off the shelf.

The evaluation card was functional the same day we received it. Using the preprogrammed PROM, we were able to load memory, execute simple programs and list the contents of memory. Our final configuration is shown in Figure 3. It consists of:

- The SC/MP microprocessor
- PROM containing the "monitor program" used to enter programs and start execution.
- RAM 512 bytes containing the programs we wrote to control data flow from one system to the other. 256 bytes are used as a line buffer.
- Input/Output interfaces to match signal levels with those on the 21MX printed circuit assemblies.

Writing programs for a microprocessor is really not much different from writing for a minicomputer, assuming one has experience with an assembler language. The interfacing, however, requires a person who has an in-depth understanding of digital logic. We were fortunate to have acquired expertise in both of these areas prior to this project.

In writing this description of the link controller, I realized that this particular approach could be used to enter data from a variety of instruments directly into a 3000. This would be especially true if only a minimal amount of real time processing of the data were required. For example, if it was necessary to log data from various temperature sensors, the sensors could be easily interfaced to the microprocessor and transferred as a record to the 3000.

As a summary, I would like to itemize the Problems which we encountered and the Solutions to those problems.

3352 GC DATA SYSTEM

P- Single User BASIC did not allow us to specify a particular terminal for input/output.

S- Use parallel interfaces defined as reader and punch.

P- Single User BASIC did not timeout on data entry from reader.

S- Use a "smart" controller which would timeout and send a special code to 3352.

3000 SOFTWARE

P- IOPENTERMDEV locked up terminals under certain situations.

S- Use this routine carefully, close terminal file properly and don't abort the program.

P- "DC1" transmitted by multiplexer.

S- Use this character to our advantage to detect ready status.

P- How do you stop a program running as a job and yet close files and preserve pointers.

S- Use a flag file which is set by a short program.

P- Passing string arrays from FORTRAN to BASIC.

S- Learn to manipulate the logical length of BASIC arrays using an SPL routine.

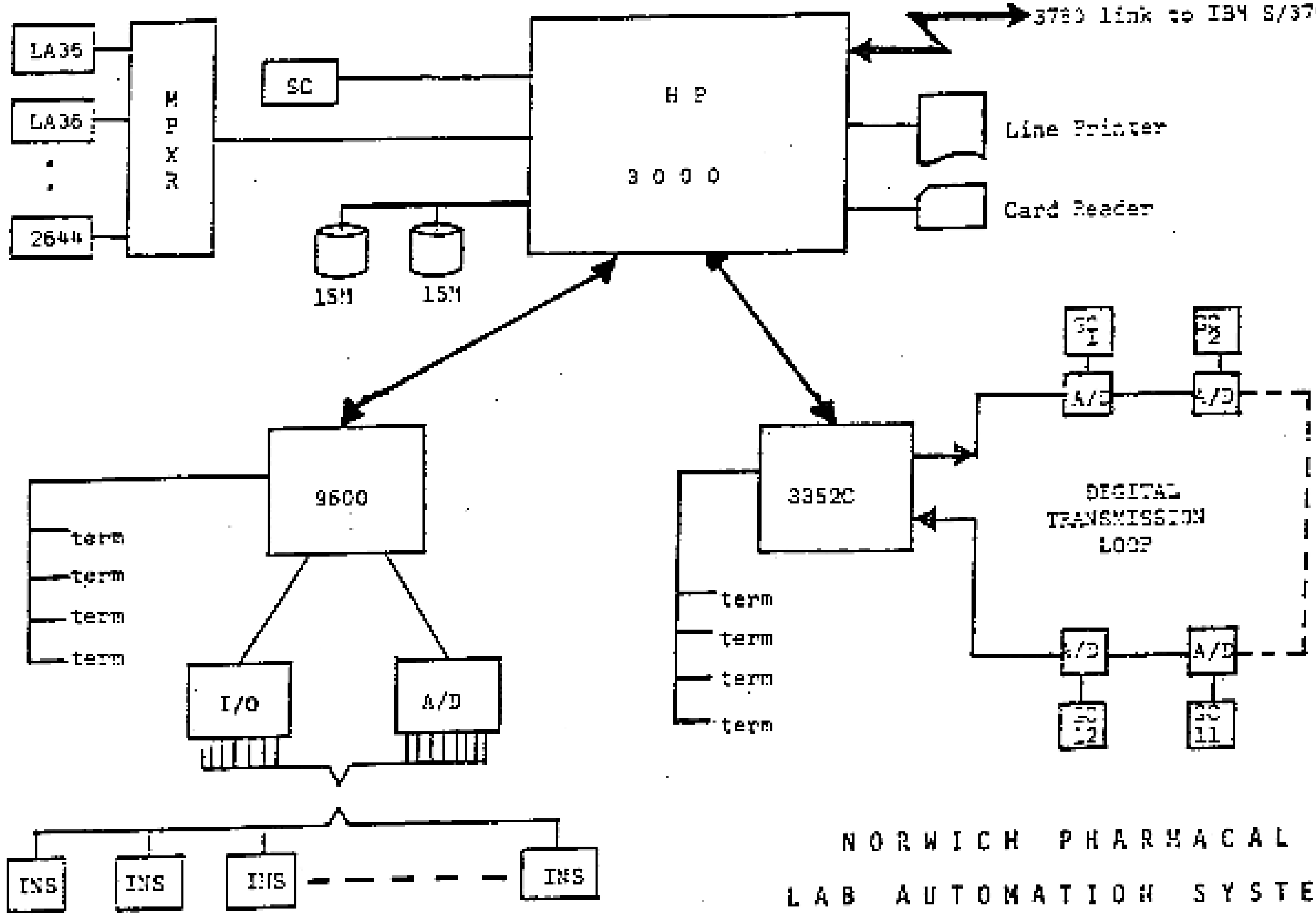
LINK CONTROLLER

P- Checking echo at 2400 Baud was difficult.

S- Analyze the echo generated by the MPXR for timing variation at different BAUD rates. (The echo is offset from the incoming pulse.)

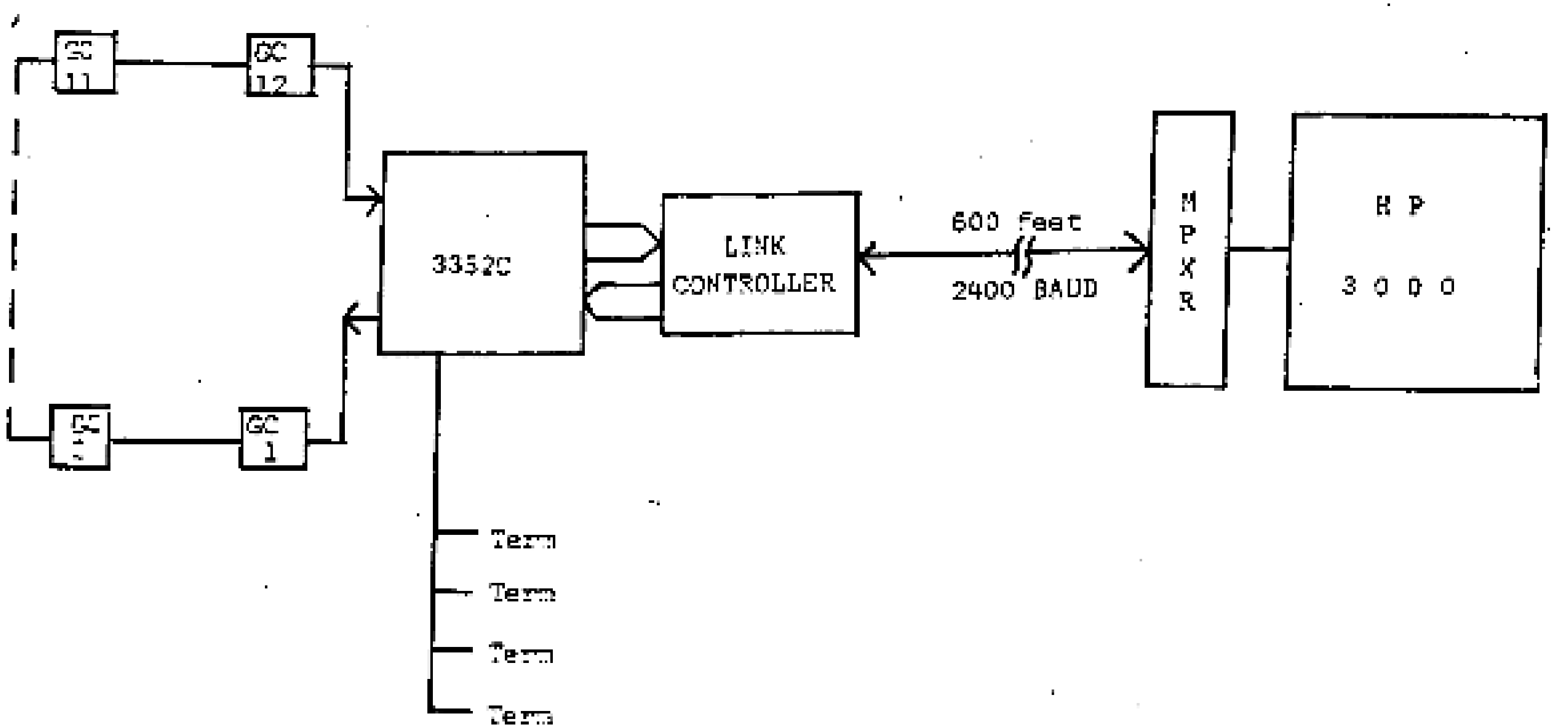
This initial success with microprocessors has encouraged us to re-examine certain applications which we might have otherwise ignored. For analytical instruments it is generally not feasible to dedicate a minicomputer to each instrument. By interfacing these instruments using microcomputers we could perform a certain amount of data reduction before sending the data to the larger computer in serial format. With the cost of micros as low as it is, I feel that a new dimension has been added to distributed processing.

FIG. 1



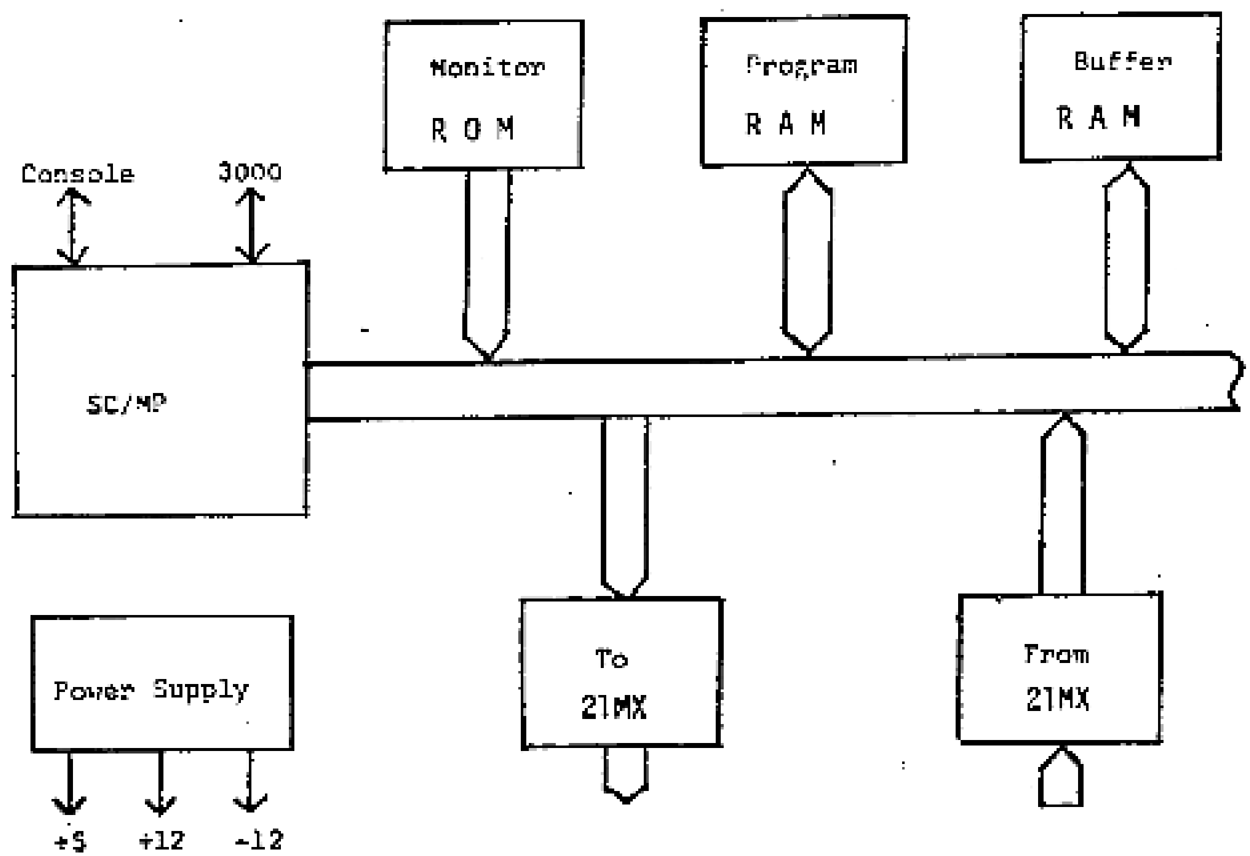
NORWICH PHARMACAL
LAB AUTOMATION SYSTEM

FIG. 2



G C LINK HARDWARE

FIG. 3



L I N K C O N T R O L L E R