Mass Spectrometers in a Time Shared Computer Environment

W. E. Reynolds, R. B. Tucker, R. A. Stillman, and J. C. Bridges
Instrumentation Research Laboratory
Genetics Department, Stanford University School of Medicine
Stanford, California

ABSTRACT

The use of a time shared computer system for mass spectrometer data acquisition and/or instrument control is described. The computers involved are an IBM 360/50 with an interconnected IBM 1800 and a classic LINC. The use described is in a time shared mode, where numerous data processing and instrumentation users are simultaneously served. The mass spectrometers connected are a mix of a high resolution double focusing, a medium resolution single focusing, and a quadrupole instrument in diverse areas, ranging to 1500 feet from the central computer. These and other instruments are served without any prior scheduling or the particular knowledge of other users. Programming is done on remote terminals in a high level language. The experiences in these modes enables comments to be made concerning the relative merits and useful roles, deficiencies, and benefits computer time sharing can offer. This work was sponsored in part by National Aeronautics and Space Administration Grant NGR-05-020-004 (Genetics Department), NIH Grant AM 04275-07-S1 (Chemistry Department), and NIH Grant 5 P07 FR00311-02 (Medical School-Computation Center, Advanced Computer for Medical Research).

In 1965 a group of Stanford researchers and computer science planners formed a committee to establish the basic design of a time-shared computer system that would not only provide the time shared terminal access as had been pioneered by MIT and others, but also would include comparable time shared data channels with connections directly into the laboratory. Early large computers and batch processing had separated the user from the computer. Time shared computers later restored the user to direct contact with the computer. Somewhat in the same manner it is hoped to free the laboratory instrumentation from the constraints of the small computer, or from the restrictions and delays of submitting recorded data to a batch system. The time shared large computer should be able to give acceptable real time service to the instrument in the laboratory.

In Figure 1 the Stanford system, ACME, is shown composed of a 360/50 with 2,065,000 bytes of core with a satellite IBM 1800. The latter functions only as low speed instrument data channel. I have included the Genetics Department's classic LINC computer in the general mass spectrometer systems. It can operate in the system or stand alone. There are three distinct instrumentation data paths possible from a laboratory to the central computer:

1. A high speed unit, up to 25 K bytes/second, called the 270X-270Y. The 270Y is remote in the laboratory. The 270X is local to the computer and has access through the multiplexer.

2. Low speed, up to 1 K samples or words per second, via the 1800 inputs.

3. Last, if used, the a-to-d converter of the LINC.

There are output channels via each path that have analogous attributes. On the left side of Figure 1 are the principal mass spectrometers of this system.

1. Double focus, High Resolution, AEI MS-9.

2. Single focus, Integer Resolution, ATLAS CH 4.

3. Quadrupole, Integer Resolution, Finnigan 1015.

4. Time-Of-Flight, Integer Resolution, Bendix Mod. 12.

A typical laboratory instrumentation center in our concept would contain a data terminal, a keyboard for programming and control, and a graphical output capability. This is functionally arranged in Figure 2.

The computer complex and domain is below the dotted line. Physically the computer connections are in the laboratory, but design, maintenance and specifications of the terminal lie with the computer administration. The special adaptors and additions

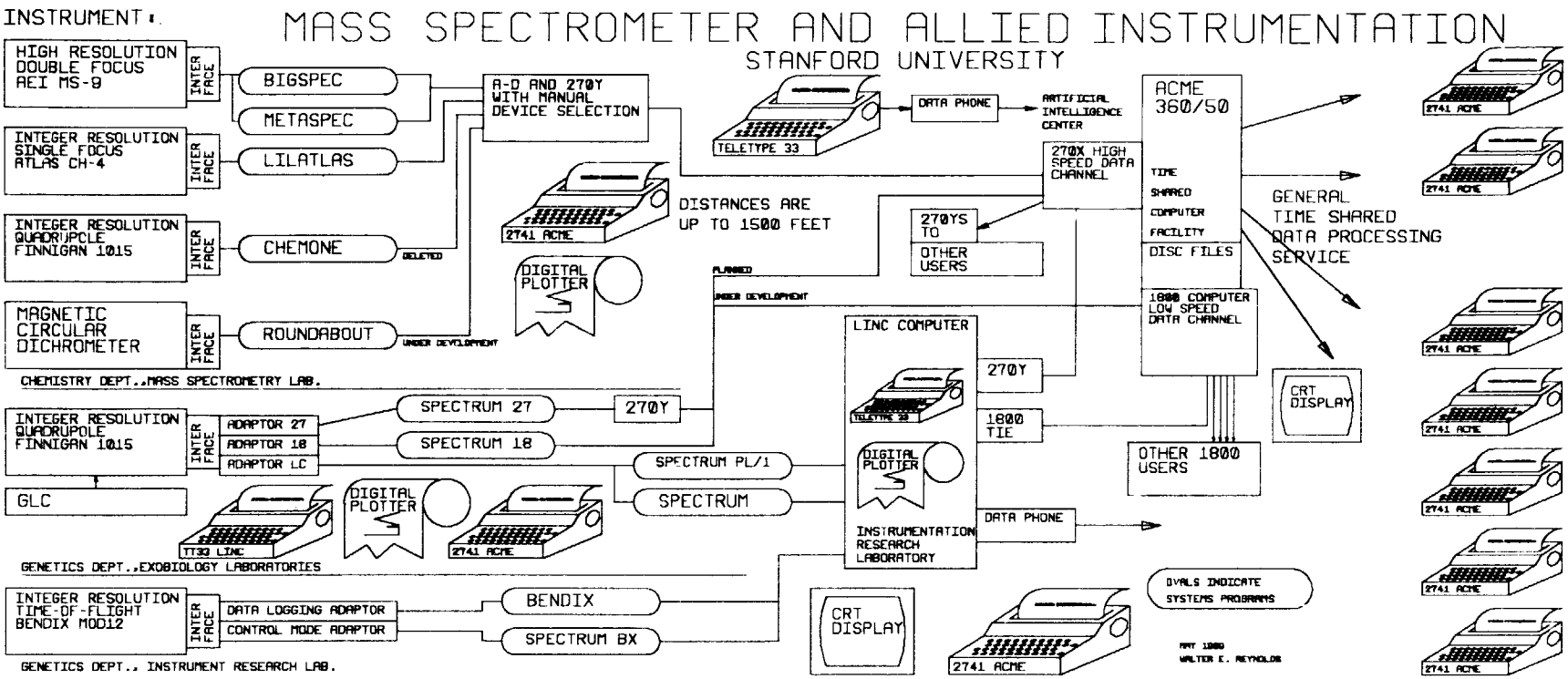


Figure 1

A diagram of the units in the mass spectrometer system employing the time-shared 360/50 computer.

needed to connect the mass spectrometer, are identifiable and indeed built, as a separate special purpose interface. Design, maintenance and even computer interchangeability are enabled or simplified by this arrangement.

Last, but not least in requirements or expense, is a system to make this work and be useful. The realization is best identified by the computer program written to effect operation. The general use concept is implied in the program. We have tended to identify these systems by their program names. In Figure 1 and Figure 2 these system names appear in the oval boxes that tie together the operational paths.

Our programming is all in PL/1. Programs may and must be written at any of the remote keyboards. Since there is an interactive compiler, programs may be modified, altered and/or controlled right at the using site at any time, even concurrently with operational use. Figure 3 shows an elementary program that would cause 100 a-to-d conversions from a laboratory instrument and then plot these values as a graph on the laboratory plotter.

The eight systems shown in Figure 1 connecting mass spectrometers of the time shared net, are of three classes. The top two, BIGSPEC and METASPEC both are quite special to the MS-9. BIGSPEC is actually a collection of three major programs for the data acquisition and processing of high resolution runs. Element mapping has not yet been included; typed output gives fractional mass position of the found peaks. Provision is being made to send this data directly to other computers at Stanford engaged in the artificial intelligence project and their organic structure elucidation programs. METASPEC is a little more than a slide rule to aid in metastable identification.

LILATLAS and CHEMONE are data logging programs and integer peak identification. This type of program has been well reported in the literature. With only minor changes to accommodate the type of scan, we will use the same program on the magnetic focus, the electrostatic quadrupole and the time-of-flight instruments.

A TIME SHARED COMPUTER INSTRUMENTED LABORATORY

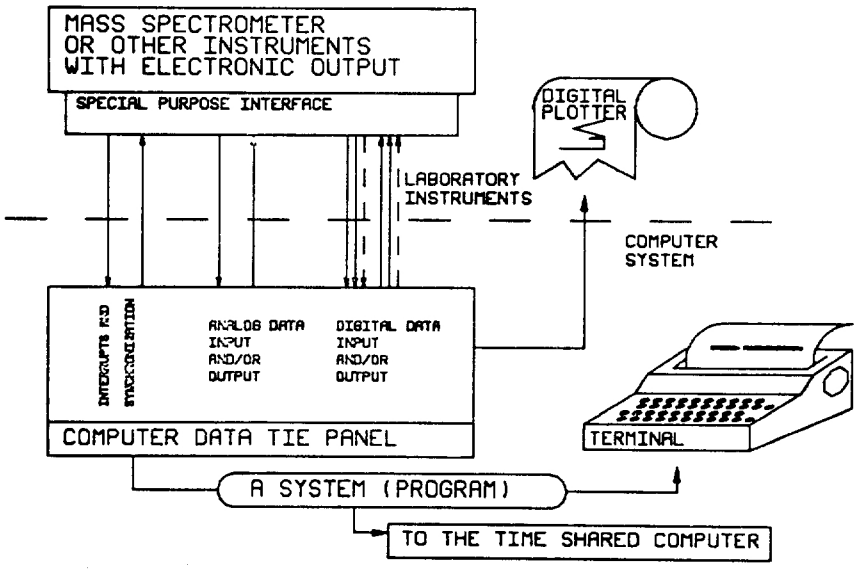


Figure 2

The SPECTRUM series are computer control and data acquisition. SPECTRUM does not use the features of the time shared system, only the LINC. The system was described at last year's E-14 in Philadelphia. The SPECTRUM logic has been rewritten into PL/1 and operates as SPECTRUM PL/1 in the time shared mode. It presently features completely automatic calibration and a mass range from 10 to 500. The other two versions of SPECTRUM will be nearly identical, but differ in the hardware path used. Interchangeability of hardware modes is made possible with the standardization of the CALL READ statement shown in the PL/1 example.

Provision is made to type or plot results in the laboratory. Figure 4 shows two of the standard plot formats we employ for integer resolution data.

An information presentation problem has arisen. Profuse data acquisition and impatience for results have led us to devise interactive data presentation methods. In the SPECTRUM systems the users typically take 10 or 20 spectra of a solid sample, and 200 or more of a GLC run. The results are in the computer and instantly available. But typing or plotting takes time and our users have become impatient with delays of more than 10 or 15 minutes.

We are now working out ways of giving data abstracts to enable the user to "home in" on the spectra or data he wants. This interactive information extraction runs like this:

a. The user asks for a certain small set of data based upon his estimate of what is the pertinent data. If his estimate is vague, presumably he will ask for a condensed set or simply indicates the set or sets.

b. The program responds with a quick presentation.

c. The user may use this computer output to improve his estimate of what he required. Return to a.

This may take the form:

a. Ask for the 8 highest peaks in all 10 spectra taken during a solid sample run.

b. Computer types the 80 items.

c. The user sees that the higher masses did not show up in the sets of 8 high peaks. But he sees that peaks from spectrum 7 onwards are saturated.

d. Reasks for the 8 highest peaks from mass 200 onwards, in spectra 1 to 6.

e. Computer responds with 48 items.

f. User notes that a good fragmentation was indicated in spectrum 4.

g. User asks for a full plot of spectrum 4.

The central time shared computer allows shared files. We are building a file system with standard formats. Each mass spectrometer user's data may be made available to all.

```
 1.000 PLOTSOME:Procedure;
10.000   /*        This is a PL/1 program to take 1000   */;
11.000   /*     sample points and then plot them.        */;
12.000   declare LINE2H (1000);
13.000   CALL READ(18, LINE2H);
14.000   /*  18 addresses the laboratory instrumentation */;
15.000   /*  terminal.                                   */;
16.000   DO I=1 to 1000;
17.000       CALL PLOT(76, I, LINE2H(I),2);
18.000   END;
19.000   /*  76 addresses a specific digital plotter     */;
20.000   /*  in the laboratory.                          */;
21.000 END PLOTSOME;

?
```

Figure 3

A sample program in PL/1 that would accept laboratory data and plot it on the laboratory plotter.

The uses of this are just emerging. Perhaps the first practical use will be that of sending samples to a different mass spectrometer laboratory. The results may be filed by the mass spectrometer operator and the requestor may get his results directly on his own laboratory terminals and process it in his own programs. All the described operations used the computer in a time shared mode and concurrent with other users and other instrumentation. No advance notice is required to operate, other than to request a "line" from the operator. The operator does reserve the right to deny service if the usage is too high. This normally happens only in the teletypewriter ports or when there is some abnormal operating condition of the computer.

Observations based upon our experience with time shared instrumentation:

A time shared large computer does greatly aid in program development. The ability to write instrumentation systems software in a high level language and with on-line compiler does enhance program development. Also since much system improvement will be largely to the software, such improvements may be introduced easier than if machine code or assembly languages had to be used.

It should be noted that much more support and computer capacity are needed for system development than for operational uses. Certainly program changes and documentation is easier with high level languages and hence less costly by an order of magnitude. The large system is peculiarly suitable and capable in this initial phase. After the algorithims are tried and proved usable and experience has been gained in the operational use, the justification of the large computer does decrease. Operations could be carried on largely or solely by a small computer.

Another useful aspect of powerful time shared compilers is the availability of common functions, exponentials, log, trigometric and the like. Also the ease of floating point and double precision does make life more enjoyable. No interaction between user and the system is enhanced. If nothing else, the output formatting ease of high level languages greatly aid programming for user interaction.

It had been our experience that small programs may be written in a tenth of the time in the time-shared mode as compared to small computer assembly languages. However the advantage diminished as the systems get larger and the structure of the high level language program gets complex.

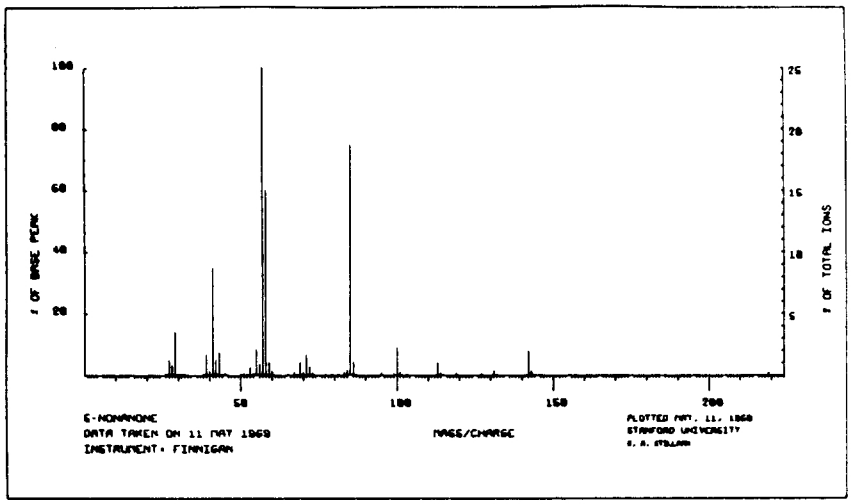
Some weaknesses that have shown up in our system:

1. The lack of overlay capability or object code stored programs or procedures can cause uneconomic or cumbersome operation. Total systems of instrument setup, data acquisition, data processing and/or filing, and data presentation can run to very large proportions. Our systems run 5 to 25 thousand words for just instructions and auxiliary working areas.

There appears to be a fundamental conflict between the incremental compiler used in very responsive time shared programming systems and precompiled program or program segments. In our system, programs are filed on disk as manuscript source code and compiled upon entry into object code. Thus a program or procedure must be recompiled each time it is brought from storage. This forces us to large programs held in core for hours even though the use is intermittent. The penalty to avoid the delays of recompiling is very uneconomical use of core. Our experience clearly indicates the need to swap programs or program segments in the order of seconds.

2. Fast disk or drum data filing should be provided. Most sophisticated time shared systems have very protective file systems that are well calculated never to make an error. But the price paid due to redundancy, flexibility, and cross checks is slow filing. Instrumentation data usually can pay a price of bit errors of the order of one in 10^9 to 10^{12} if speed is gained. And there is the real time need to file data rapidly to avoid large core storage. Again the price paid for slow filing is the large amount of core needed to buffer real time data input.

3. For instrument-computer interaction, fast channel turn around is desirable. This is the ability to do sequential operations of input, then output or vice versa. This is not necessarily allowed with complex computer operating systems. Such systems tend to be oriented to fast data streams in one direction as to or from disk or tape units. If one wished to accept a datum point, and then react with an output, then repeat, it may be found that it takes a big computer many milliseconds to change from input to output and back. Similar difficulties may be found if complex orders of data input/output are attempted.



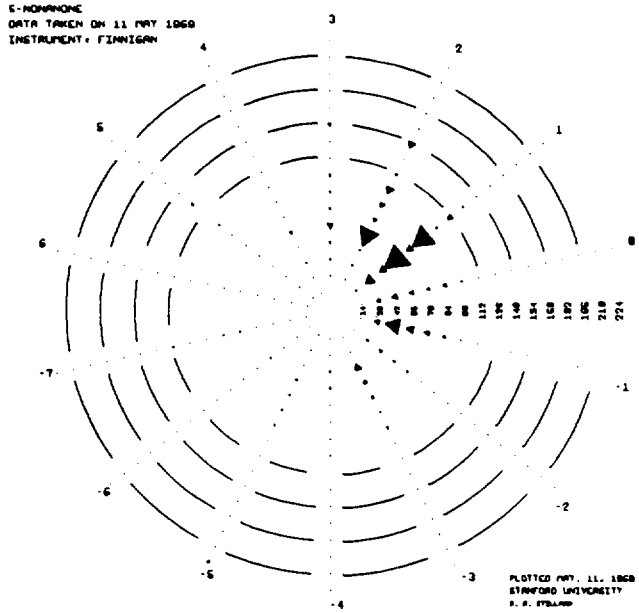


Figure 4

4. At this state of the art large computer systems are prone to "crashes". The complex program that makes a large computer a time shared facility will be faced with conditions or malfunctions it cannot handle. At such times the system becomes inoperative and remains so until operator intervention restores normal operation. Often data or program extras made sometime prior to the "crash" are lost. Also the user's programs normally have to be recompiled. Even a few moments of master computer down time can cost the user hours of his and his laboratory's time.

Despite the limitations described, which largely are the price paid for a developmental system, we feel that the mass spectrometer complex we have is now reasonably current with the state of the art in mass spectrometer instrumentation. But this is not the point of merit. The fact that the system is capable of much development is the most exciting feature. The basic instrumentation approach, the data channels provided, the versatile access to the computer from the laboratory, the general purpose computer service in real time, and high degree of programmability provided, all give us a vehicle especially suitable for continuing the enhancement of mass spectrometer-computer-instrumentation.

INTERACTIVE INFORMATION PRESENTATION, EDITING, AND FILING

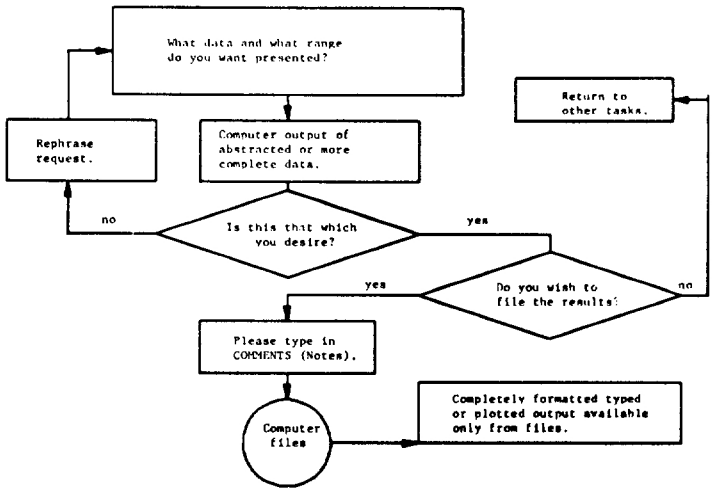


Figure 5

REFERENCES

1. W. J. Sanders, G. Breitbard, G. Wiederhold, et al., "An Advanced Computer for Medical Research", Fall Joint Computer Conference Proceedings, ACM, page 497, 1967.

2. J. F. Corbato, et al., "The Compatible Time-Sharing System", MIT Press, 1963.

3. R. J. Spinard, "Automation in the laboratory", Science 158 (3797), 55, 1967.

4. R. A. Hites and K. Biemann, "Computer recording and processing of low resolution mass spectra", International Mass Spectrometry Conference, Berlin, Sept. 1967.

5. W. E. Reynolds, J. Bridges and T. Coburn, "A computer operated mass spectrometer", Genetics Dept., Instrum. Research Lab. Technical Report IRL 1062.

6. C. A. McDowell, Ed., Mass Spectrometry, McGraw-Hill, New York, 1963.

7. B. Halpern, J. W. Westley, E. C. Levinthal, and J. Lederberg, "The Pasteur Probe: An Assay for Molecular Asymmetry", Life Sciences and Space Research IV, M. Florkin and A. Dollfus, Eds., p. 239-249, North-Holland, Amsterdam, 1967.

8. B. Halpern, J. W. Westley, I. von Wredenhagen, and J. Lederberg, "Optical Resolution of DL amino acids by gas chromatography and mass spectrometry", Biochem. Biophys. Res. Comm., 20, 710, 1965.

9. W. E. Reynolds, J. C. Bridges, R. B. Tucker and T. B. Coburn, "Computer control of mass analyzers", Conference Proceedings of Sixteenth Annual Conference on Mass Spectrometry, ASTM Committee E-14, Pittsburgh, Pa., 1968.

Dist: Staff/All