Set II

BOLT BERANEK AND NEWMAN INC

CONSULTING · DEVELOPMENT · RESEARCH

Report No. 1784 12 May 1969

SPECIFICATION OF AN IMPROVED DATA PROCESSING SYSTEM FOR BIO-SCIENCE LABORATORIES

Alexander A. McKenzie
Philip La Follette

Prepared for:

Bio-Science Laboratories
7600 Tyrone Avenue
Van Nuys, California

Report No. 1784

12 May 1969

SPECIFICATION OF AN IMPROVED DATA PROCESSING SYSTEM FOR BIO-SCIENCE LABORATORIES

Alexander A. McKenzie
Philip La Follette

Prepared for:

Bio-Science Laboratories
7600 Tyrone Avenue
Van Nuys, California

TABLE OF CONTENTS

INTRODUCTION

This report contains the results and conclusions of the System Specification phase of a program for the development of a comprehensive computer system for Bio-Science Laboratories, Inc. (BSL). This effort was performed by Bolt Beranek and Newman Inc. (BBN) under the terms of the contract reproduced in Appendix I. The specification and design was conducted by A. McKenzie, P. La Follette, S. Ornstein, F. Heart, and P. Castleman of the Systems Development Department in Cambridge with the assistance of other BBN personnel from Cambridge and Van Nuys.

The design of the computer system described in this report is based on a previous study conducted by BBN for BSL and summarized in BBN Report No. 1766, "Recommendations for an Improved Data Processing Facility for Bio-Science Laboratories". A familiarity with that report is assumed in this document. There are, of course, a number of differences between our present conception of the BSL computer system and the ideas presented in our earlier report; the most important of these differences relates to the entry procedure as discussed in Sections 1 and 6 of this report. Our present conception of entry room procedure is based on a BSL internal document prepared by Mr. Keller, Dr. Winkelman, and Dr. Thiers entitled "Proposed Entry System for Bio-Science Laboratories" which is reproduced in Appendix II.

This report first discusses the programs which reside in memory in the real-time computer (Sections 1 and 2). All important functions to be performed by these programs are described, in order that BSL personnel can judge the completeness of the real-time system design. Based on this program design, core memory requirements are estimated in Section 3.

Section 4 contains a brief discussion of our recommendations for result transmission by wire service. The conclusions presented in this discussion are somewhat different from those presented in our first report.

Those functions which are to be performed by batch processes are described briefly in Section 5. As with the description of the real-time programs, all important functions to be performed by the batch programs are mentioned.

Sections 6 and 7 deal with the hardware configuration which BBN recommends. The first of these sections describes the function of each component, while Section 7 addresses the costs of various methods of hardware procurement and maintenance.

Finally, Section 8 discusses the problems involved in introducing the system into full-scale operation, presents recommendations designed to minimize these problems, and discusses the issues of operating staff briefly.

1. ENTRY AND RESULT POSTING

Since the design of the real-time system is closely dependent on the processes of entry and result posting, these two real-time processes are summarized at the outset.

Specimen Entry

The general process of entry, including box-opening, typing, labeling, splits, checking, etc. is assumed to follow the outline of the BSL document in Appendix II. We here summarize only the interactive typing process.

The typist inputs either an account number or a client's name. In the former case, the account number is checked for validity. In the latter case, the client's account number is looked up and if present reported to the typist. There may be more than one client with the same name, and so the system should allow the operator to confirm or reject the client number found, on the basis of such secondary information as the client's address.

Test requests are input either by test code number or abbreviated test name. In either case, the input is checked for validity. If a test code number is input, the system could respond with the abbreviated name of the test.

If the typist indicates that the patient is to be billed, the system checks for entries already in the patient billing file which have the same client number and name (surname). If any such matches are found, the typist is asked to specify whether any of these is in fact the same patient as the one whose specimen is being entered. If not, a new entry is made in the patient billing file for this patient.

If a test is one to which considerations of 24-hour volume are appropriate, the operator types in this information. (The system knows when to expect 24-hour volume data.)

All other information typed in at entry time (e.g., patient age) is input without special checking by the system.

Result Posting

The general treatment of result posting was covered in BBN Report #1766. Here we summarize the interactive typing process.

There are 3 types of result posting procedures.

1) Accepting manual posting – The operator may enter either a tray number or a test number. If a tray number is entered, the system will respond by typing out the position and specimen number for each specimen and waiting for the result. If a test number is entered, the system will accept a series of specimen number-result value pairs.

2) LINC posting – The LINC should be expected to communicate at Teletype speeds. It should be expected to output a tray number and a set of results, including pools, specimens, and standards. It also will send "flag"

characters with some results. If results for an entire tray are received with no flags, then the results should be immediately transferred from the pending file to the specimen file. If any flags are received for the entire batch, the result values should all be left in the pending file.

3) Posting from pending file – Results stored in the pending file by the LINC must eventually be posted manually. In order to facilitate this process, the user should be able to enter a test number and obtain a list of all trays in the pending file for that test, or enter a tray number and obtain a list of all the results (and flags) for that tray, along with the corresponding specimen numbers. The user should then have the facility to specify a set of ranges of specimens which should be posted. The ranges should probably consist of tray position numbers and can be specified as individual numbers or as pairs of numbers separated by a dash. When the user signifies that there are no more pending results from this tray which should be posted, the program should probably delete all entries for this tray from the pending file.

There are some features which should be present in all posting schemes. First, the Tray Position Index should contain all entries for a single tray in a set of records reserved for that tray. The tray position index entries should all be deleted under either of two conditions:

1) All specimens in the tray have had results posted (automatically or manually).

2) Results were entered into the pending file by the LINC with some flags, and the manual procedure for posting some or all of these results has been performed.

At the time the tray entries are deleted, the tray number may again become available for assignment.

Second, many of the tests permit results of more than one set of units. For this reason, the terminal user must be permitted to specify units as well as a numerical result. This probably does not apply to results transmitted by the LINC.

Third, it will sometimes be desirable to enter result values as "internal remarks" rather than as results. This is likely to occur each time a result value is far outside the normal range for the test, since in these cases the tests are frequently repeated. The laboratory would, however, like a record of the first result, so that subsequent results can be compared with it.

Fourth, the laboratory personnel will frequently wish to add one or more of the standard "notes" to the report which is to be sent to the client. Thus, the result posting programs must permit the users to add the codes for one or more notes to the test results when necessary.

Fifth, some tests call for more than one numerical result, that is, the results are assays of a variety of different substances. The result posting program must allow for this fact, and should assist in the posting procedure by outputting the "names" of the values which are expected for these multiple-result tests.

Sixth, various statistics will be kept on the values of pool results, and perhaps also on certain specimen results. One of these statistics will be designed to detect trends in the results. If a trend of steadily increasing (or decreasing) values for the pool specimens is detected, the system should produce a warning message for the laboratory supervisor or the quality control director.

2. OTHER REAL-TIME PROGRAMS

In addition to the entry and result posting processes already described, a wide variety of other processes must be carried out by real-time programs. Each of these processes is described briefly here.

1) Change Specimen records – After a specimen has been processed through the entry procedure it will sometimes become necessary to alter the information about the specimen. The only information which is certain to remain unchanged is the specimen number. Accordingly, a real-time program must be provided which accepts a specimen number and permits any of the other information about the specimen to be changed. This may also involve altering one or more of the indexes to the specimen file.

2) Listings – As described in our earlier report there are a number of listings which will be required for laboratory operation. These should provide for counts of all specimens waiting to be processed in a given laboratory, or lists of the specimen numbers *and tray positions* for either specific tests or all tests assigned to a particular laboratory. Examples of the type of information available from the listing programs are shown on pages 10-14 of our earlier report.

3) Worksheets – It must be possible for the laboratory personnel to generate worksheets for the various tests which are performed. There should almost certainly be two alternative methods of specifying the specimens which

should appear on the worksheet; these are by tray number and by specimen number. Worksheets should contain all information about the specimen, as well as a work area specialized for the particular test. This specialized work area may consist of a number of blank lines or it may contain text; in any case the format of the work area will be specified by the BSL technical personnel and a program to alter this format will be supplied.

4) Inquiries – Pages 18-22 and 34 of our earlier report discuss a number of inquiries which might be instituted from the laboratory. In addition to these, there are a number of other inquiry types which must be provided for users of the system. Additional types of inquiry will be:

a) A list of all trays of pending results for a particular test code. The user will supply the test code and the program will respond with a list of all the trays containing pending results for that test.

b) A list of all results in a particular tray in the pending file. In this case the user will enter a tray number and the program will respond with a list of all the specimen numbers and their associated results and flags.

c) In conjunction with payment posting, the financial office personnel require the ability to search the patient billing file by patient name. The retrieval program, given a patient name, will produce a list of all patients with that name, the associated street address, and a list of the numbers of all invoices for that patient.

d) One of the functions of the result posting programs will be to maintain an up-to-date set of statistics on the results of each type of test. These statistics will be printed as a batch report at various intervals. At least some of these statistics, however, should be available to the inquiry program. It may be desirable to omit some statistics from real-time inquiry; for example the standard deviation might be omitted if the variance were available in order to avoid including a square-root routine in the real-time system.

5) Payment posting - The real-time system will provide for posting payments from both clients and patients. In addition, the batch system will include a program for posting patient payments when the payment is exact and the prepunched invoice card is returned. As noted above, inquiry functions will permit the accounting personnel to "look up" the invoice numbers of patient payments when the invoice is not returned, or to find the account number of a client who has sent a payment without including his bill stub. Therefore, the payment posting programs will only accept either account numbers or invoice numbers for identification of accounts to be credited. The programs must format the posted payments in a way that will permit easy visual comparison of the terminal typescript and an adding machine tape, and must also provide a total of all payments posted from each terminal. These two functions will be used to cross-check payments entered into the computer system and the adding machine tapes which accompany bank deposit slips.

We expect the system to maintain a file of all unpaid patient billing in the files accessible through the accounting office terminals. This will permit the accounting personnel to respond to telephone inquiries about the status of patient payments. If a record of a patient billing does not appear in this file, it may be presumed to have been paid in full.

The programs which are used to process patient payments may also be required to cancel a patient billing and substitute billing to an insurance company. BSL frequently does not know, until the patient has been billed, that the bill should be covered by insurance. Thus, there must be a way for the accounting personnel, upon receipt of insurance information from the patient, to stop further bills from being sent to the patient and instead initiate billing to the insurer.

For patient payment programs, there must be a method for marking an account "paid in full" even though the actual dollar amount is not full payment. This is because some medical plans apparently have fixed maxima for certain standard tests, which are lower than BSL rates. BSL seems to not bill the patient for the remainder theoretically due.

6) Cancellations – A program must be available to users at all terminal locations to order cancellation of particular tests. This program must deduct the appropriate amount from client billing files and reduce all appropriate test counts, as well as marking the test cancelled. There are many possible reasons for cancellation and the laboratory management requires the ability to determine the frequency

of each of these. Accordingly, there will be a small file of cancellation codes which must also be updated by the cancellation program.

7) Tray reassignments – During the entry process, each specimen (including the "additional" specimens generated by splitting) is assigned to a particular tray position. This tray position assignment is used for two purposes, to assist the laboratory personnel in locating a particular specimen and, in the case of tests which are performed in large batches, to simplify the result posting procedure. There are several circumstances which will make it necessary for the laboratory personnel to move specimens from one batch (i.e., tray position) to another, and if this is done the system must be informed of the moves in order to handle posting correctly. Therefore, there must be a tray assignment program which will accept from the user information about the "present" and intended tray locations of a specimen and restructure the files to reflect this change. The "present" location information may be either a tray location or a specimen number-test code pair. There may also be a requirement to create new batches in the laboratory. For this purpose, the tray reassignment program will probably expect a complete list of either tray locations or specimen number-test code pairs for all of the specimens to be assigned to the new batch. It may, however, prove more desirable to handle these special batches entirely through the result posting procedure.

8) Master file updates – There are a number of files which will require periodic updating. The programs to perform the updating functions may all be real-time

programs, but it is more likely that some of them will be implemented as batch processes. The update programs must have the capability to update:

a) The client file. Clients must be added or deleted, addresses must be changed, the number of copies of reports required must be changed. Also included in this general program is the ability to change container inventory levels or place special orders for additional containers (N.B., the entry program also permits container orders.)

b) The test master file. Test charges may be changed, the worksheet text can be altered, and tests may be added to or deleted from the set of tests performed at BSL.

c) The test name thesaurus. We hope that it will be possible to include a test name thesaurus in the system; this would permit the entry personnel to enter a test name, rather than a test code number, and have the computer respond with the correct test code. If this feature is included there must be a mechanism for adding test names (and the associated codes) to the thesaurus.

9) Service routines – There are several service routines which, although never seen directly by the system's users, are vital to the functioning of the real-time system. These include the File Service Routine, the Terminal Service Routine, and the Common Arithmetic Module, all of which are described in Appendix III.

3. CORE MEMORY REQUIREMENTS

In order to obtain reliability and backup, and for other reasons, BBN has specified a dual computer system where the two machines are identical. In the batch system, program modules may be stored on discs, and only occupy core space when used. However, to maintain adequate response time, the majority of the real time programs must reside in core at all times. Thus, the core requirements for the real time system are controlling, and we summarize those requirements.

For the various parts of the real-time system software described in the preceding sections, we estimate the approximate core memory requirements as follows:

File service routine	2500 words
Terminal service routine	1000 words
Common arithmetic routines	1000 words
Specimen-entry program	1500 words
Result-posting program (including LINC service)	1500 words
All other real-time programs (estimated at 500 words for each program described in Section 2, except 1000 words for Inquiries)	4500 words
Total real-time program size	12000 words

In addition, the following buffers will be needed:

Table of disc cylinder availability (four bits for each cylinder)	200 words
Two 123-word system disc buffers for each disc unit	700 words
Buffers for top-level block of each index	1500 words
8 label printing buffers	300 words
One of each of the following for each of 26 terminals and for LINC and console Teletypewriter service:	
123-word disc buffer	3400 words
40 words of temporary storage	1100 words
40-word terminal buffer	1100 words
Total Buffer Size	8500 words
Program Size	12000 words
Total Core Memory Requirement	20500 words

This estimate is not exact, and some margin must be provided for error. We now recommend core memory size of 24,576 words, providing a margin of about 20%.

4. WIRE SERVICE

We have examined the various possible methods for arranging wire service outputs from the new Bio-Science on-line computer facility. First, continued use of TELEX is appropriate, rather than other types of wire service, such as WATS or TWX. Second, we still believe that attempts should be made to expand the use of output wire service with due attention to economy.

The BBN Phase I report implied the use of wire service on an "instantaneous" basis as soon as a result is posted by a laboratory. We have reconsidered this issue in light of our discussions at BSL and in light of a more careful examination of the costs of wire service operation. We now believe that wire service outputs should indeed be batched rather than sent on an instantaneous basis. We analyzed data taken in February 1969 and attempted to estimate the extra transmission costs that would be involved for different batching periods. The following table summarizes these conclusions (based upon a once per day batching period taken to be unity).

BATCH PERIOD	EXTRA TRANSMISSION COSTS
Once per day	0
Once per eight hours	8%
Once per four hours	26%
Instantaneously	80%

Based upon this examination we would recommend that wire service batching should be done two or three times per 24 hour period, thus keeping the extra cost well under 10%.

In view of this recommendation to batch the wire service outputs we also reconsidered the issue of whether an automatic dialing unit is appropriate. Our conclusion is still that we recommend an automatic dialing facility. First, we believe that the costs are roughly the same. Second, we believe that lower error rates and more orderly procedures are possible with the automatic dialing unit. Third, we are not presently suggesting the inclusion of a card punch in the system and this means that the present method of batching wire service outputs would not be possible.

Finally, we feel that the automatic dialing unit should provide a sounder base for expanding wire service outputs than the present manual procedures using cards. We recommend that the automatic dialing unit be a Western Union Telex type #12150 which includes a model #32 teleprinter and has an approximate total monthly charge of $103.00 per month.

Since the wire service outputs will be batched we feel that this TELEX interface should be initially attached to the batch computer system. An attachment to the computer system requires a computer interface and our current estimate of the purchase price for this interface is $3525.

5. BATCH PROCESSES

A large number of functions which should be performed by the computer system can best be handled as batch processes. Figure 1 is a flowchart illustrating the interaction of programs in the batch and real-time systems. Communication between the two systems will be by means of the disc files only, although both computers will also have access to the magnetic tapes. We believe that most of the batch programs should be written in FORTRAN, an English-like language which is available for use with all computers suitable for the BSL system.

There are a number of modifications which must be made to the FORTRAN language processor if this implementation method is followed; in particular the FORTRAN input-output routines must be modified to be able to handle the disc file record format which will be used by the real-time programs. Additional functions must also be added, probably in the form of library subroutines, to simplify the processing of alphabetic text. Also, the batch operating system must be slightly modified to prevent the batch system from monopolizing either the disc files or the tape units and thus stalling the real-time programs. Once these changes are made, however, the job of writing, debugging, and documenting the batch programs should be greatly speeded up. Furthermore, it should be much less costly to make changes in the batch programs if they are written in FORTRAN than would be the case if they were written in assembly language.

Several of the batch programs listed below produce reports for internal use at BSL. It should be noted that certain aspects of the file design for the system are dependent on the frequency

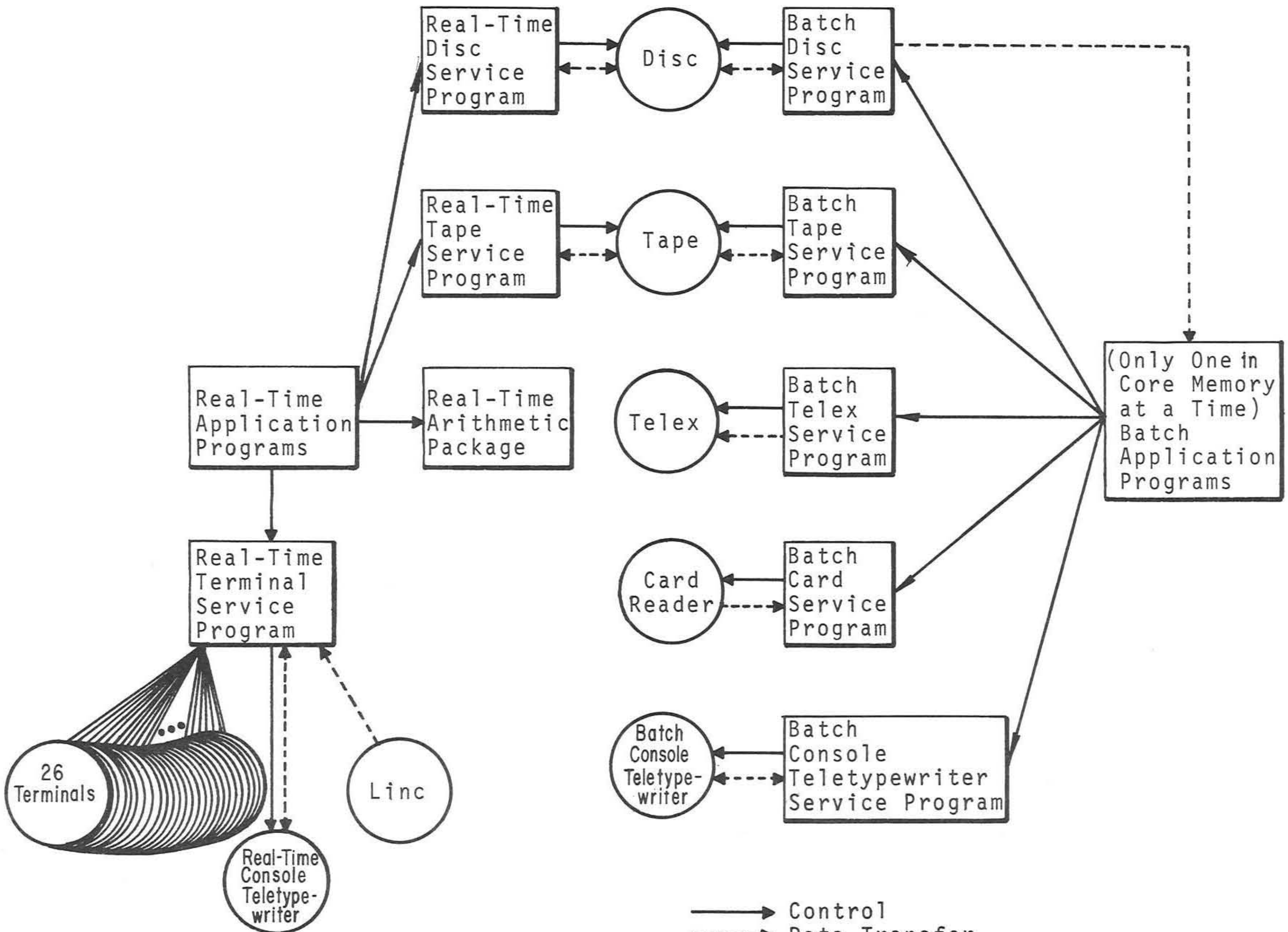


FIGURE 1
INTERACTION OF PROGRAMS IN THE BATCH AND REAL-TIME SYSTEMS

with which these programs are to be run, or, more precisely, the number of different time intervals which should be the basis for reporting. For example, if a total is to be accumulated for one week, reported, and then reset to zero in preparation for the next week's report only one storage unit is required in the file design for this total. On the other hand, the total might be desired on both a weekly and a quarterly basis; in this case an additional cell would be required to store the "current-quarter-to-date" total which would only be reset to zero after the quarterly report was produced. For this reason BSL should be prepared to specify the desired frequency of production of each report quite early in the implementation phase.

There will also be times when it is desirable to produce a batch report before the end of the time period for which data is normally accumulated for that report. For this reason, the function of resetting accumulated totals to zero will always be an option which must be specified by the computer operator at the time when the batch program is run.

Some, perhaps most, of the batch programs must be capable of producing only partial output based on a set of input parameters. A simple example of this type of program is result report printing; it may be desirable to print all reports going to a certain set of ZIP codes, while not printing any other reports. In some of these cases the desired parameter set may be so large, yet so frequently used, that it is most reasonable to submit the parameters to the batch system via a punched card deck rather than through the operator's console. This is not to imply, however, that the system programming design will rely on the massive use of punched card decks.

The batch programs which will be required for system operation are listed below, together with the types of options which will be included. We believe that each of the separate batch programs, together with its I/O device handling subroutines and arithmetic and text processing subroutines, should fit easily into the 24,576-word core memory we have recommended.

1) FORTRAN Language Processor – The arithmetic capabilities of the FORTRAN language should be available for batch process use by BSL statisticians. The language processor will be supplied with the computer by the manufacturer at no additional charge.

2) Result Printing – This program will group all reports for a single client except those which require attachments; reports requiring attachments will be printed after all other reports in a given run, again grouped by client. The program will include the capability of printing all of the notes (i.e., "attachments"), which consist exclusively of text, directly on the report forms, as well as recognizing those notes which actually must be attached by hand. There may also be a requirement to compute the probability that the test result is "normal"; this requirement is currently under study by BSL. The report program will be designed to take input from the disc files; however, if it should prove economically feasible the program will be designed to produce selected reports from long term (magnetic tape) storage. In addition the program must have the capability of printing only a subset of the reports based on parameters such as the client ZIP code

(first three digits, for geographic area separation) or the number of copies of a report required by the client (some clients require multiple copies of each report).

3) Request Slip Preparation and Container Inventory – This function includes preparation of mailing labels and request slips on both a regular and a "special" basis. The program must identify the type (blood, urine, miscellaneous) of request slip to be prepared, as well as identifying the types of containers to be shipped.

4) Patient and Client Billing – Client billing is performed on a monthly basis. A completely itemized invoice must be prepared for each client, based on specimens submitted (rather than results reported) during the month. Each client master record will contain a number of parameters applying to the billing of that client; these parameters include

a) Number of invoice copies required

b) Whether the unpaid previous balance should be shown

c) Any applicable across-the-board discount percentage

d) Whether a TELEX discount should be applied.

There may also be a requirement for generating a duplicate of a previously prepared invoice. Although the system can be programmed to store invoice copies for an indefinite length of time on magnetic tape for this purpose, we believe that a better solution might be the production of a carbon copy of each invoice for temporary storage in the accounting department. The choice between these methods can be made by BSL without affecting system programming costs.

Patient billing is currently done more than once a month. There must be provision for preparing first, second, and third notices in separate runs. We believe that the current billing procedure (i.e., printing first notices on prepunched card-stock invoices) should be retained, since the majority of patients billed return the first notice cards with their payments. A payment-posting batch program will be developed to process these cards; the program will produce a totaled listing of credits to be compared with an adding machine tape as a cross check on actual income.

Some patient billings are converted to insurance billings. We understand that it may be desirable to develop programs for preparing the appropriate insurance forms. We have <ins>not</ins> included the costs of such development in our estimates of the batch programming costs. Although we would be happy to develop programs of this type if specifically requested by BSL, we believe that BSL programmers could probably perform this task more economically after installation of the system.

5) Review of Client Billing — Every so often (the most logical time being very shortly before the client billing run) there should be a review of client information by a batch program. This review will provide several measures of BSL business as options, including the following:

a) A percentage figure may be supplied as a parameter. The name, address, outstanding balance, and recent payments of any client who has remitted less than that percentage of the balance due at time of previous billing will be printed.

b) There are one or more codes assigned to each client. These indicate the type of business in which the client is engaged (i.e., hospital, clinic, private physician, and in the case of physicians, their specialties). The review should produce totals of either dollar volume or specimen volume submitted by each type of client.

c) Client ZIP code will be stored with the client record. The review program should use the code to produce totals of either dollar volume or specimen volume submitted by geographic area.

d) A dollar amount may be supplied as a parameter. The name and address of any client whose current monthly charges exceed that amount will be printed. This information may also be grouped by geographic area.

One object in the design of this program will be to limit the number of complete passes over the client file to one, if at all possible. This is desirable in order to limit the interference of the batch system with the real-time system in disc access. However, it should be noted that several reports are expected from one run of this program. This may be accomplished by use of magnetic tape for temporary storage.

6) Programs Related to Payroll Data — We recommend maintaining all payroll related data on magnetic tape, thus making all of it completely unavailable to the real-time system in order to insure privacy. Input to the payroll programs should probably be supplied via punched cards. Payroll related functions include:

a) Updating of employee status information including name, address, sex, marital status, date of birth, education, employee number, classification, department, hire date, shift, social security number, union or non-union, technical or non-technical, credit union member or not.

b) Production of the "personnel list" each month. This list includes all of the employee status information listed above.

c) Production of paychecks (or a payroll list) every two weeks. This function should include provision for overtime payments, unpaid leave time, etc. For this function, and many of the other payroll functions, existing program logic can undoubtedly be used.

d) Preparation of a monthly payroll report.

e) Preparation of a quarterly payroll report including a calculation of taxable earnings.

f) Production of W2 forms and possibly state tax forms each January.

g) Preparation of the "labor analysis" report each pay period.

h) Preparation of a "laboratory efficiency" index for each laboratory each pay period from the following formula:

$$\text{Lab efficiency} = \frac{\text{Actual \$ income/technician hour}}{\text{Goal \$ income/technician hour}}$$

i) Determination of the "overtime index" for all BSL employees according to the following formula:

$$\text{Overtime index} = \frac{\text{Total overtime salary payments}}{\text{Total base salary payments}}$$

7) Result Statistics – A program will be provided to produce a report of statistics related to each test. Although these statistics are likely to consist primarily of the mean, standard deviation, variance and other measures of the results of testing pool specimens, we are aware that BSL is currently studying the utility of accumulating statistics on the results of specimens submitted for some or all tests. If these statistics are accumulated, they will also be reported by this program.

8) Speed of Reporting – An existing program compiles a list of all results posted on a selected day. The entry date for each posted specimen is determined, and a tabulation is printed showing number of specimens entered on each day (<u>and</u> posted today), number of working days since entry, overall number of results posted, average number of working days from entry to posting, and percentage of all results "overdue". The number and percentage of cancelled tests for each entry day (and the totals) are

also shown. With the new system it seems reasonable to accept some loss of detail in return for a complete report on speed of reporting for the entire time period. We would suggest that the "speed of reporting" report contain the following information.

Test number

Test name

Total tests reported during the period

Total tests cancelled during the period (including those entered during an earlier period).

Percentage of tests cancelled, i.e.,

$$\frac{\text{Total cancelled}}{\text{Total reported}} \times 100$$

(note that this is not a true percentage)

Average time in lab for all specimens posted during the period

Percentage of all results posted which are overdue.

The report will break down the reporting by department, section, and test number as is done presently. It should be noted that this report will also serve as a "test volume" report.

9) New Test Income - Income data can be accumulated for all tests performed. This information might then be printed in report form by a batch program. It should be noted that this is not exactly the same as the cost times

the number of tests, due to discounts, research prices, etc. Nevertheless, we recommend that the test volume information be used, and that this report be eliminated from the system.

10) Cancellations — A count will be kept of the number of tests cancelled for each of the possible reasons. This information may be retrieved in summary form whenever necessary.

11) "Note" Frequency – A count will be kept of the number of each of the types of notes used during result reporting. This information may be retrieved in summary form whenever necessary.

12) Advertising – Bulk mailing lists will be stored either on punched cards or on magnetic tape. There must be a program to produce mailing labels from these lists. Also, if the lists are stored on magnetic tape, it may be desirable to update them.

13) TELEX Service – A program to transmit by TELEX service all results accumulated since the last TELEX transmission.

14) Long-Term Storage Update – A program to delete specimen information from the disc, including all indexes, and store it on magnetic tape.

15) Long-Term Storage Retrieval – A program to search tape storage using at least the parameters of account number, patient name, and test code.

16) Disc Dump - A program to transfer all information from the working discs to backup discs (possibly to magnetic tape). This program will stall the real-time system and must therefore be run at periods of low system use.

17) Emergency Disc Reload - If one or more discs are found to be in error, it will be necessary to reload by updating a backup disc from the journal file. This function will be required to index all of the input data, and for this reason it may be desirable to perform this function through the real-time system, which will contain the most elaborate disc-handling program. The emergency reload can be an overlay to either system, since the real-time system cannot be functioning when reloading is required.

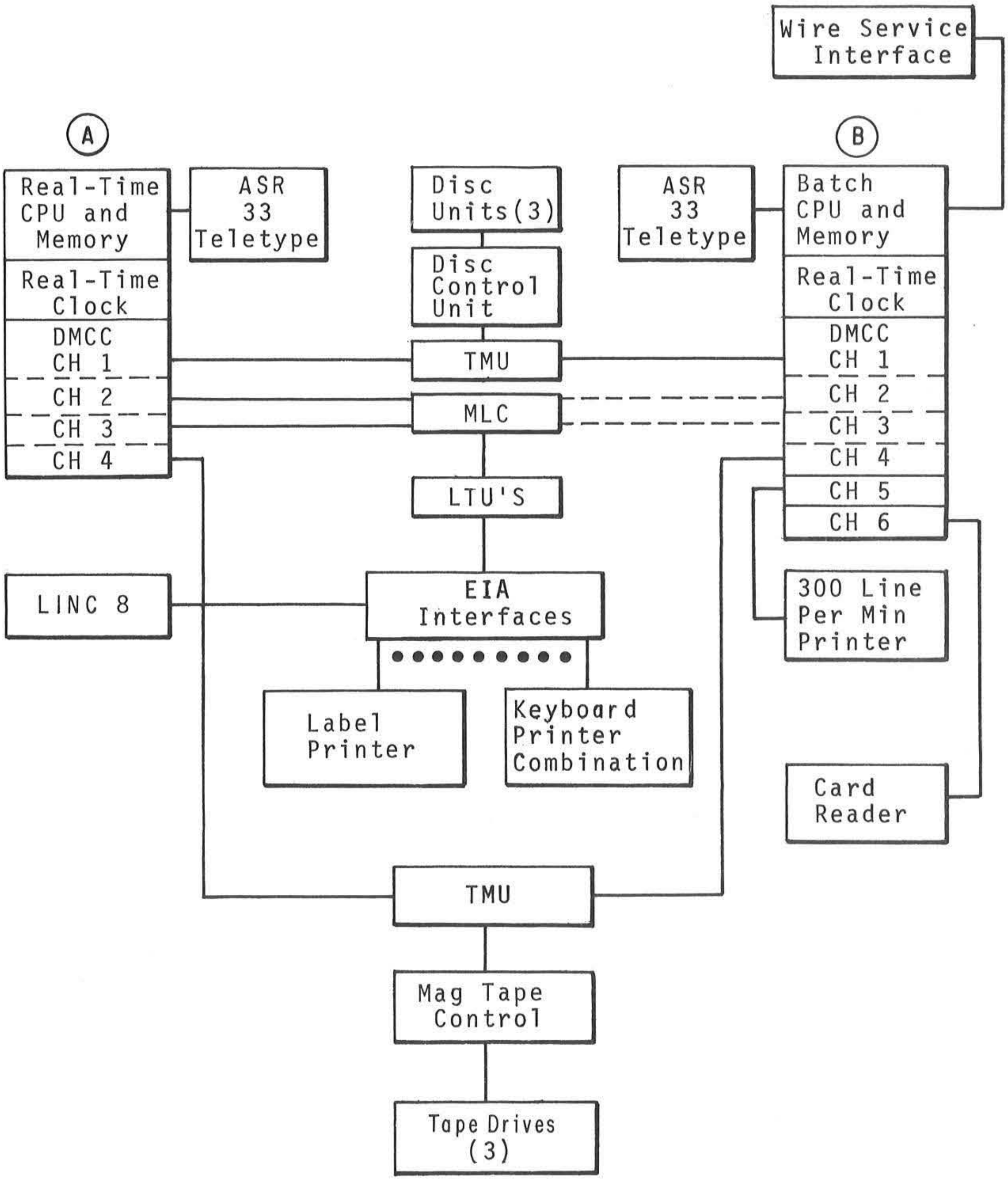
18) Disc Reorganization - As old information is released from disc storage and placed on magnetic tape, space becomes available on the disc for re-use. Because of the possible interactions of real-time programs, however, this space cannot be immediately re-used. Instead, a reorganization program must be run to add this space to a "file" of free space, from which the real-time programs then can obtain records as necessary.

6. SYSTEM HARDWARE

Figure 2 is a block diagram of the recommended system. It is essentially the dual processor system outlined in our earlier report. We considered in detail three possible choices for the computer: The Digital Equipment Corporation PDP-9, the SDS Sigma 2, and the Honeywell DDP-516. Other machines were considered briefly and discarded for various reasons. Of the three primary contenders we finally settled on the Honeywell machine based on our own experience with the machine (and with the PDP-9) as well as on the experience of others which generally indicates the 516 to be well designed and reliable. The same cannot be said of the PDP-9. Delay in availability of disc storage for the Sigma 2 decided us against that machine.

Either computer is capable of performing the real time function. The Wire Service Interface is connected to computer B which normally performs this job. The Multi-Line-Controller is normally connected to A, but simple recabling (a ten minute job) permits it to be reconnected to B (which is normally the batch machine) in the event of a failure or routine maintenance on machine A. This switches all of the terminals and the LINC to B. The line printer and card reader stay connected to B, but are not used by the real-time programs. The disc and tape subsystems are always connected to both machines which share them via TMU's (Time Multiplex Units).

The DMCC (Data Multiplex Communication Control) unit in each computer provides for Input/Output to be performed in block transfers without involving the program on a character-by-character or word-by-word basis. Basically this reduces the



SYSTEM BLOCK DIAGRAM

FIGURE 2

amount of the computer time which is used to perform simple Input/Output. A DMCC can be expanded to handle up to 16 sub-channels, arranged in a priority servicing order. The highest priority channel in each machine is assigned to the disc system. When a disc transfer is actually in progress a word is transferred approximately every 12 microseconds (μs), each word requiring 4 machine cycles, i.e., 4 μs. This uses 33% of the machine capacity during the actual transfer. Of course even if the disc is being referenced very frequently, much of the time is spent in positioning the proper information and only a tiny fraction of the time is spent actually transferring information; thus the actual percentage usage of the machine capacity is much less.

Since the disc units are shared between the two machines one of the machines may move the head of a unit unbeknownst to the other machine. Since positioning from track to track is performed on a relative basis (e.g., "go forward three tracks"), when a disc unit is shared a special head address register is required to communicate the actual head position from one computer to the other. Three such Absolute Address Registers, one for each disc unit, appear in the equipment list although they do not appear separately on the System Block Diagram.

The disc system utilizes three rather than the originally suggested four disc drives. However, each drive stores about 12 million characters on 20 surfaces for a total of about 36 million characters. This is a new drive (Honeywell EDP model 273) and requires a modification to the controller to permit it to handle 20 surface drives (as opposed to the usual 10 surfaces). The disc system thus has a long delivery time and will likely be the critical item in the timing of the delivery of the final system.

The Multi-Line-Controller is connected to DMCC channels 2 & 3 — normally in machine A but able to be recabled to machine B as discussed above. This unit deals with all of the terminals including the LINC. It contains the common logic and a large delay line buffer for assembling and storing incoming and outgoing characters for each line. In addition, each line also requires some separate individual logic for storing individual bits as they come in or as they are to go out. These units are known as Line Termination Units (LTU's). Finally each line must be electrically adjusted to standard specifications ("EIA") in order to work with the terminals. The terminals are discussed separately below.

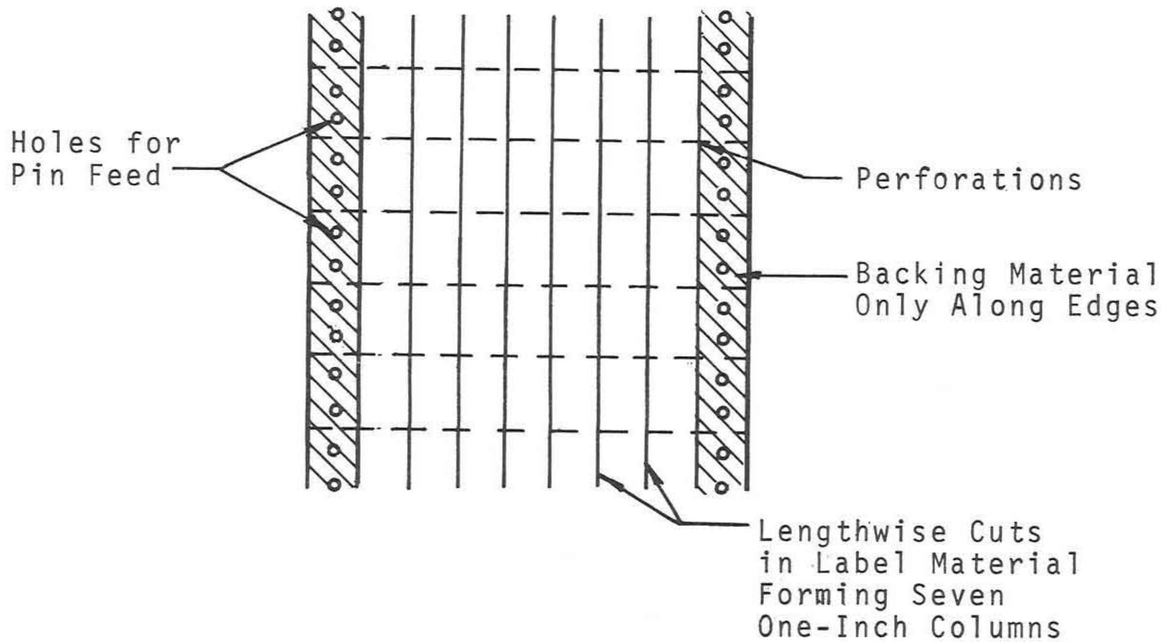
Three tapes are shared by the two computers through a second TMU connected to DMCC subchannel 4 in each computer. The tape drives we have recommended are 36 inch per second transports capable of either 200 or 556 bits per inch. These are the least expensive units with prices increasing somewhat for higher bit density (capacity) and considerably for higher speed (i.e., faster access).

This rounds out the DMCC subchannels for machine A. Machine B, for its normal batch processing role, however, has two furthur DMCC subchannels. The first is for a high-speed line printer — 300 lines per minute, up to 120 columns of printing, accepting paper of various widths. The second is for a 200 card/minute card reader. The card punch was eliminated from the system simply because no genuine need was found for it.

We have investigated a number of terminal configurations both for the normal interactive laboratory and office locations and for the more specialized problems of the entry room. Our choice of terminal for all locations except the entry room is a combination of a thermal-effect printer manufactured by the National Cash Register Company (NCR) and Keyboard manufactured by the Microswitch Division of Honeywell.

The Honeywell Keyboard Model 64SW must be modified for computer communications use. These modifications require a one-time engineering cost of $3,000. The Keyboard units may then be obtained in any quantity at a cost of $945 each. The NCR printer (EM-T2, Model B) is capable of operation at 30 characters per second, a rate which we feel is well matched to the laboratory requirements. The printer is virtually silent and is capable of printing 96 different characters. We recommend planning for 18 of these interactive terminals, allocated as follows: Laboratories (10), Office (2), Accounting (2), Executive Wing (1), Computer Room (1), Mail Room (1), and the Internal "branch" laboratory (1).

The terminals for the entry process are used both for the entry of information by the operator and for the printing of labels. Model 37 Teletypes are used for the entry terminals inasmuch as they are capable of accepting label material while the NCR printers are not. The paper used in the terminal is a roll of label stock with a line of perforations across the roll every 1.5". These perforations go through both the label surface material and the backing. In addition, the label surface material has continuous cuts along the length of the roll with 1 inch spacing. This is shown in Figure 3.



LABEL PAPER FOR SPECIMEN ENTRY

FIGURE 3

After typing the input (including any commentary on the input by the computer), the operator waits while the computer types out the requisite labels. The Model 37 Teletype operates at 15 characters per second and thus the operator would have approximately a seven second wait after completing his input while the computer types labels for a typical two-test case (or approximately two seconds more for each additional test). Since the Model 37 operates at a 150 bit/second rate as opposed to the 300 bit/second terminal units in the laboratory, we recommend a flexible interface combination (Model 670 A, B, C) which is capable of handling units of up to four different speeds and can thus service both the laboratory terminals and the entry terminals. In addition, because of its multi-speed capability it can also service the 110 bit/second line to the LINC.

The latest version of the Model 37 Teletype, which would be used as the entry terminal, is equipped with a sprocket paper feed mechanism and a "reverse line feed" feature. These features make it possible for the system to return to a line which has already been partially typed. We would therefore expect the entry process to be performed as follows:

1) The entry typist will type all input information (including possible computer interaction) in a relatively free format. The computer will keep track of the length of the longest typed line.

2) After the typist has completed the input operation, the system will output all labels required for the specimen. If there is sufficient space remaining in the row of labels where the input was performed, the output will be placed there; otherwise the output will be placed on the next label row.

3) At the conclusion of output, the system will space the paper forward several lines, until the row of perforations below the labels is in a convenient position for separation.

4) The typist will tear off the labels (and the associated input typing) to be placed in the specimen tray. When this operation is completed the typist will signal the computer by typing a special character. Upon recognition of this character, the system will retract the paper (by means of the "reverse line feed" feature) to the correct position for the next input operation.

Although the operations described above appear to be simple and practical, it is important to note that we are not completely confident that the system can be put into use as described. We are still investigating two potential problem areas, which are:

1) Will the typist be able to easily separate the label material at the perforations, without ripping or otherwise damaging the labels?

2) Will the Teletype require mechanical modification in order to retract the label material far enough for use, and subsequently advance the paper without jamming?

Some prototype label material is being manufactured to BBN specifications so that a simulated operation can be carried out; unfortunately this material is not yet completed.

Although we believe that the probability that the system described above will perform satisfactorily is quite high, we have described an alternative solution to the entry room problems in Appendix IV. We do not recommend this alternate solution because of its considerably higher cost, but we are convinced that it is a technically feasible alternative. In addition, Appendix IV discusses an alternative to the NCR Printer and Microswitch Keyboard arrangement which could be used if unusual problems were encountered in the delivery or reliability of the recommended configuration.

7. EQUIPMENT COSTS

The required hardware is summarized in Table 1. The price figures are taken directly from manufacturer quotations, and the quotations themselves are given in Appendix V.

There are three ways for BSL to acquire the computer equipment. These are

1) Rental of standard items from Honeywell and purchase of special items,
2) "Full payout" leasing from a third-party leasing company, and
3) Direct purchase.

In our view, rental plans are appropriate when a system is likely to be upgraded within a few years, or when there is risk of not wanting the system at all in a year or two. We feel that the recommended system has enough growth potential to last a long time within the reasonably anticipated growth of BSL. We also see little chance that BSL will wish to eliminate its use of an on-line system once it exists. Thus, we do not recommend rental of the equipment under any circumstances. Honeywell's most favorable rental rates are included in Appendix V under the heading "Monthly Lease Price, 4 year". Note (in Appendix V) that a purchase option does exist in this rental plan, but it is a fairly "expensive" option.

"Full payout" leasing is essentially a capitalization method for financing large purchases. Under a typical plan, a third party would purchase the equipment specified by BSL and lease the equipment to BSL at a monthly rate of perhaps about $21 per $1000 of purchase price over a five year period. At the conclusion of this period, BSL could purchase the equipment for a small percent of the original purchase price (about 5%). BSL would, under this type of arrangement, be responsible for maintenance, taxes, and other such expenses related to the equipment.

Based on these considerations, BBN recommends direct purchase if the capital is available. We also recommend that BSL obtain a maintenance contract covering the equipment supplied by Honeywell; the costs for such a maintenance contract are included in Table 1. The terms of such a maintenance contract, and the costs of extensions to the contract are shown in Appendix V.

There is a related purchasing issue, having to do with the potential involvement of BBN or other system implementation organization. BBN would initially install the system on its premises, program and debug it and then deliver it to BSL. Often (although not necessarily), when BBN delivers a system in this fashion, BBN would actually purchase the hardware and then deliver the system. The advantage of such an arrangement is that BBN knows <u>how</u> to purchase such complex hardware; the issues of acceptance testing, debugging of special hardware items, checking out manufacturer-furnished software, and "dickering" with suppliers is in fact a difficult area at best. In addition, BBN, or other such organizations, may occasionally be able to obtain a better price, or a discount, on some major items. On the other hand, direct purchase, leasing, or rental by BSL is a simpler contractual arrangement and BBN might still assist in acceptance test arrangements as a BSL agent. We wish to point out that the prices shown in Table 1 are direct manufacturer list prices, and so not reflect anticipated purchase discounts or BBN charges.

In addition to the equipment listed, there are two options to be considered.

1) Line Printer — A 650 line per minute printer (Model 516-702A) is available instead of the 300 line per minute unit at a cost of $32,000 ($8,000 higher than the model we recommend). The extra speed did not appear necessary to us but is an option open to BSL.

2) Additional terminal units should probably be on hand as substitutes in the event of terminal breakdown. Experience indicates that electro-mechanical terminal units are prone to breakdown occasionally. Furthermore, the entry room Teletypes and the NCR printers will not be acquired through Honeywell and will therefore not be covered under the recommended maintenance contract. We therefore recommend that at least one spare terminal unit of each type be purchased to avoid disturbing operations in the event of terminal trouble. This would mean buying an extra Model 37 Teletype, an NCR Thermal printer and a Microswitch Keyboard.

TABLE 1

Qty	Model No.	Description	Purchase Price: Unit Price	Purchase Price: Total Price	Maintenance On-Call Monthly
Central Computers					
2	516-05	DDP-516 general purpose digital computer with 24,576 words of core memory	$63,800.	$127,600.	$300.
2	516-12	Real Time Clock	1,500.	3,000.	20.
2	516-25	Priority Interrupt	1,600.	3,200.	10.
2	516-25-1	4 Additional Priority Interrupt lines	400.	800.	N/C
2	516-53	ASR-33 Teletype unit	1,200.	2,400.	100.
2	516-360A	Data Multiplexed Communication Control (DMCC)	4,190.	8,380.	30.
1	516-9001	I/O cabinet with power supply	2,200.	2,200.	N/C
		TOTAL CENTRAL COMPUTERS		147,580.	460.
Card Reader					
1	516-61	Card Reader, 200 cpm	8,900.	8,900.	40.
1	N/A	DMC subchannel	500.	500.	N/C
		TOTAL CARD READER		9,400.	40.
Line Printer					
1	516-7050	Line Printer, 120 columns, 300 lpm	24,000.	24,000.	165.
1	N/A	DMC subchannel	500.	500.	N/C
		TOTAL LINE PRINTER		24,500	165.

TABLE 1 (continued)

Qty	Model No.	Description	Purchase Price: Unit Price	Purchase Price: Total Price	Maintenance On-Call Monthly
Disc Subsystem					
1	516-300A	Time Miltiplex Unit (TMU)	$5,900.	$5,900.	$20.
3	516-300B	Absolute Address Register for TMU	1,325.	3,975.	10.
1	516-4600	Moving head disc file controller capable of handling up to four disc storage units	11,400.	11,400.	60.
1	N/A	Modification to 4600 for 273 discs	7,720.	7,720.	N/C
3	N/A	EDP model 273 Disc drives (20 surfaces)	31,500.	94,500.	420.
2	516-4606	DMC subchannel	500.	1,000.	N/C
		TOTAL DISC SUBSYSTEM		124,495.	510.
Tape Subsystem					
1	516-300A	Time Multiplex Unit (TMU)	5,900.	5,900.	20.
1	516-4100	Magnetic tape control unit, controls up to 4 transports	10,130.	10,130.	50.
2	516-4106	DMC subchannel	500.	1,000.	N/C
3	516-4130	36 ips Magnetic Tape Transport, 200/556 bpi	13,225.	39,675.	240.
		TOTAL TAPE SUBSYSTEM		56,705.	310.
Wire Service Interface					
1	516-6110	DLC	2,900.	2,900.	15.
1	516-6180	Auto Call Unit	625.	625.	3.
		TOTAL WIRE SERVICE		3,525.	18.

TABLE 1 (continued)

Qty	Model No.	Description	Purchase Price: Unit Price	Purchase Price: Total Price	Maintenance On-Call Monthly
Terminal Subsystem					
1	516-670A	Low Capacity Multi-Line Controller (LC-MLC) for 32 lines	$12,550.	$12,550	$40.
1	516-670B	Line Termination Unit (LTU) for 32 FDX lines	17,400.	17,400.	55.
1	516-670C	EIA Interface for 32 FDX lines	3,120.	3,120.	10.
1	516-670G	Multicode Mix	395.	395.	N/C
2	516-670H	Multispeed Mix	120.	240.	N/C
1	N/A	Microswitch Terminal Key-board	3,945.	3,945.	10.
17	N/A	Additional Terminal Unit (Microswitch Keyboard)	945.	16,065.	85.
18	EM-T2	NCR Thermal Printer	1,750.	31,500	N/A
8	KSR-37	Teletype (pin feed)	2,230	17,840	N/A
		TOTAL TERMINAL SUBSYSTEM		103,055.	200.
		TOTAL SYSTEM		$469,260.	$1,703.

8. SYSTEM INTRODUCTION

The introduction of a new computer system is potentially quite disruptive and care must be exercised to minimize such disruption.

The first group of issues concern physical facilities. Space must be available to install, test, and operate the new system in parallel with the present batch system. Bio-Science must expect at least three months overlap in the two systems after installation. Cabling must be installed prior to system installation and space must be made available for laboratory and office terminals. The entry room represents a special problem; it would be most desirable to construct a new entry room without disrupting current entry operations. If this is not possible, revision of the entry room will have to be delayed until the system is about ready to go on-line.

The second group of issues concern training. After system installation and during debugging and trial operation, the entry personnel will have to be trained in use of the new terminal. Lab personnel will have to be trained and given opportunities for trial use of the system. We view this training as occurring at BSL just after system installation but well before the system is on-line.

The third group of issues concerns phaseover. As we have pointed out before, the Bio-Science system does have separable functions which may be modified in series. This is very important to graceful phaseover.

It is crucial to Bio-Science that result reports that leave the building are correct. Thus considerable checking and review are in order before the products of the new system are put in the mail. We suggest the following staged turn-on:

Stage 1 – Simulation and Debugging

After installation, all parts of the system will be exercised using test input, special simulation tests, etc. When it is believed that the system works, checkout should proceed to Stage 2.

Stage 2 – Overlapped Entry

The entry room terminals will be used to enter data in parallel with the standard entry process. Thus, request slips will still proceed as currently, and the entire lab, result reporting, and batch system will go on as usual. Programmers and BSL personnel will monitor the input process, check on the new system's handling of the data, obtain test samples of work sheet requests, etc., and generally determine whether the system is properly handling the inputs. This overlap operation may require some temporary personnel in the entry room for a few weeks. Only when testing shows satisfactory behavior should BSL enter Stage 3.

Stage 3 – Overlapped Entry and Result Reporting

To the process of Stage 2, we suggest adding result reporting from the labs, in parallel with normal result reporting on cards. We believe that this load will not be much greater than the load required of the labs for full system operation, since writing a result on a card versus a worksheet before terminal entry should not be a major difference. Perhaps in the very high volume labs this parallel operation may be difficult and temporary help might be needed. At this point, very complete testing of the system can occur, with sample outputs printed, etc. (All this still maintains the card-oriented result reporting and the new system has not yet mailed a report). At this point BSL should send in trial samples, ask for results, queries, and so forth, to convince themselves that the system quality is adequate.

Stage 4 – On-Line

Finally, outputs from the system should be used, and the card operation phased out.

The final group of problems concerns personnel allocation, retraining, hiring, and dismissal. We feel that BSL must expect some extra personnel charges for a few months while the system is being introduced. It is quite difficult to state details about the transition period without close interaction with BSL.

Our views about steady state personnel issues are in agreement with the BSL estimates contained in the BSL internal document "Proposed Entry System for Bio-Science Laboratories" and with our earlier views. Thus we believe that the entire existing entry and Keypunch operation will be transformed into the following staffing structure:

Purpose	No. Personnel
Open box — Remove specimen and request slip	4
Keyboard entry	8
Affix labels	4
Split	2
Check table	1
Supervisor	1
Keypunch Operator	1
TOTAL	21-23

During the time that the system is in operation a computer operator should be available. The operator will be fairly busy during times when the batch system is being used heavily, but will primarily be performing a "baby-sitting" function at other times. Therefore, although we estimate that the equivalent of approximately five full-time operators will be required, we expect that at least two of these operators will be available to perform other functions during most of their working hours.

We do not expect the installation of the system to have a significant effect on the required office personnel. We should note, however, that office personnel will no longer be required to perform TELEX-related functions, and that the process of adding "attachments" should be greatly simplified.

It is still quite difficult to assess the requirements for additional clerical personnel in the individual laboratories. Certainly the Thyroid and Automation laboratories will require substantial clerical time for such activities as result posting; but it is not clear that this will require additional staffing, since clerical personnel are already assigned to these laboratories. It is not unreasonable, however, to assume that one or two more clerks will be required for these two laboratories. The other laboratories, with significantly lower work loads, are much less likely to require additional personnel since a clerk is already assigned to each of them. Thus it seems reasonable to predict a maximum of five additional clerical personnel required for laboratory operations.

APPENDIX III - SERVICE ROUTINES

A. File Service Routine

In view of the central role of the disc file in the operations of the system, we have examined in some detail the functions to be performed by the disc-handling software module in the real-time system, and how they might be implemented. The design given here is a way in which the various disc functions could be effected.

Index Structure

The indexes provide the means for looking up a segment of the data base (e.g., a specimen-record) by one of its attributes (e.g., the client who submitted it). An index, then, will contain, for every "key" (the value of the attribute used as a criterion) occurring in the data base, a list of all pieces of data filed under that key. In many cases it is also useful to include some additional identifying information for each piece of data, which is not logically necessary for the indexing function but which helps to avoid disc accesses to unwanted data.

An index has a multiple-level tree structure in which the bottom level is organized differently from the other levels. The bottom level is a linear chain of disc blocks containing, for each key, a list of entries filed under that key, each (possibly) with its own additional identification. Each block in the next higher level contains pointers to the blocks on the next lower level, along with the keys that appear first in those blocks. Thus the process of looking up an index entry consists of tracing down the tree from the top level, which is a single disc block, to the bottom level. Time-consuming disc accesses can be saved if the top-level block of each index is kept in the core memory of the real-time system as well as on the disc.

When an entry is added to an index, it is inserted into the proper place in the bottom-level chain. If there is not room, in the disc block into which it is to be inserted, for the new entry, that disc block is split into two blocks. In this case, there is now a new block in the bottom-level chain, so that (unless the new block begins with the same key as the one before it) a new entry must be made in the next higher level of the index. The addition of an entry at this level may in turn necessitate splitting a block and inserting a new entry in the next higher level; thus changes may be propagated all the way up to the top level of the index. When an entry is deleted from an index, a similar procedure is followed to update all levels which have to be updated.

Index Block Structure: Bottom Level

```
     Pointer to next block in chain (disc address), or
       "last block" flag
     Pointer to previous block in chain (disc address), or
       "first block" flag
  ┌─ Pointer to next key in this block
  │
  │  Key
  │ {Further identifying information (if any)
  │  Pointer to data base element (disc address)
  │
  │ {Further identifying information
  │  Pointer to data base element (disc address)
  │            .
  │            .
  │            .
  │            .
  │ {Further identifying information
  │  Pointer to data base element (disc address)
  │
  └→ Pointer to next key in this block
  ┌─ Key
  │ {Further identifying information
  │  Pointer to data base element
  │            .
  │            .
  │            .
  │            .
  └→ Last-key-in-this-block flag
     Key
    {Further identifying information
     Pointer to data base element
               .
               .
               .
               .
    {Further identifying information
     Pointer to data base element
```

Index Block Structure: All But Bottom Level

Number of entries in this block

{Key
Pointer (disc address) to first block on next lower level having this key as its first entry

{Key
Pointer (disc address)

.
.
.
.

{Key
Pointer

Disc Operations

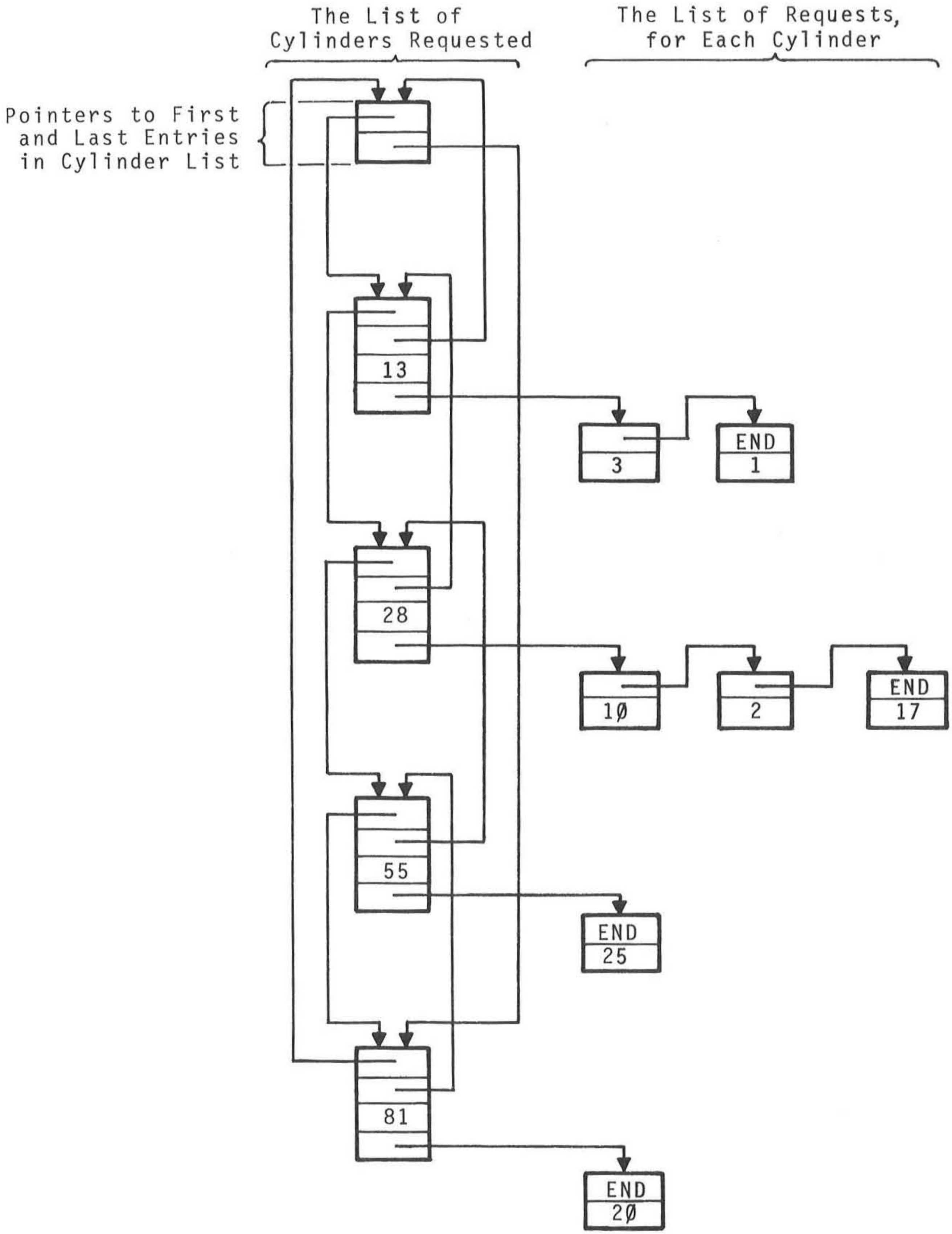
The disc-handler program services several different kinds of requests for disc operations. The functions available are:

1) Read a block from a disc, given its address.
2) Rewrite an already-existing block, given its address and the new contents of the block.
3) Create a new block on the disc in an unused area, given the contents of the block. The information is also written on a "journal" file which is maintained on magnetic tape.
4) Index lookup. There are two cases:
   a) Find the first entry in a specified index for a given key.
   b) Find the next entry in the same index for the same key.
5) Add an entry to a specified index, given a key and a disc address (and, sometimes, additional identifying information to be put into the index with the disc address).
6) Delete an entry from a specified index, given a key and a disc address. (The entry is not physically removed from the index but is marked with a special code to indicate that it has been deleted.)
7) Modify the additional identifying information stored in a specified index under a given key, given the disc address associated with the identifying information to be modified.

The Disc Queue

Since more than one request for disc service may be pending at one time, the disc handler must have a way to defer requests and to decide in what order to process them. For each disc drive in the system, there is a list (queue) of all the requests for disc operations on that drive that have been made to, but not yet serviced by, the disc handler. Each time a disc operation is completed, the entry for that operation is removed from the list on which it appears, and the disc service routine chooses from that list the next request to be processed. Only one request is processed at a time for each drive.

The queue for each drive has two structural levels. On the first level, it is a list of all head positions (cylinders) for which there are operations requested, ordered by head position. For each cylinder in the first-level chain, there is a subsidiary list of all operations requested for that cylinder, in chronological order according to the time each request was made. An example of a typical disc queue is shown on the following page.



SAMPLE DISC QUEUE

In this example, processes 3 and 1 have made requests for service on cylinder 13; process 10, 2, and 17 for cylinder 28; process 25 for cylinder 55; and process 20 for cylinder 81.

The purpose of this structure is to expedite the task of deciding which request for a drive is to be honored next, each time the drive becomes available as the result of the completion of a previous operation. At this time, the disc handler deletes the queue entry for the request just processed. If the queue still has any requests in it, the next request to be processed is chosen as follows: if there is another request pending on the same cylinder as the one just satisfied, that request is processed next. If there is no request waiting for the same cylinder but there are requests waiting for other cylinders, the first request on the next higher numbered cylinder (or next lower numbered cylinder depending on whether the current search direction is inward or outward) is chosen; if there are no cylinders requested in the current search direction, the search direction is reversed and the first cylinder found in the new search direction is chosen. Thus, if the queue is as in the above example, and if the request of process number 10 is being serviced, and if the current search direction is inward (toward the higher-numbered cylinders), and if no additional entries are placed on the queue, the requests now waiting will be answered in the order of 2, 17, 25, 20, 3, 1.

B. Terminal Service Routine

For each terminal there is a buffer with room for about 80 characters, which is used for communication between the program controlling the terminal and the terminal service routines. Here are placed messages to be sent to the terminal, as well as messages being input from the terminal. The program controlling the terminal has the ability to inhibit the use of the buffer by the terminal service routines, so that the buffer may be used for other purposes while there is no terminal activity in progress.

Since there are no interrupts generated to signal the completion of the output of a character, the terminal-output interrupt activity will have to be started up by a clock whose rate is chosen in accordance with the character rate of the fastest device attached to the Multi-Line Controller. The terminal-output interrupt program will check for lines on which output has been completed since the last terminal-output interrupt, and for each such line initiate the transmission of the next character of output. When the last character of output to a terminal has been sent, the terminal service routines will change the status of the terminal from "waiting for terminal output" to "ready to run".

The terminal-input routine, which, unlike the output routine, can be initiated by an interrupt, will serve the functions of echoing the character received from a keyboard back to the printer associated with that keyboard; storing the character in the terminal buffer for the appropriate terminal; and recognizing when input has been completed and changing the status of the terminal from "waiting for terminal input" to "ready to run".

If a character is input from a terminal on which there is no pending request for input, the character is ignored and not echoed; the typist can thus see that his input is not appropriate.

C. Common Arithmetic Module

Because of the wide range of possible values for test results, and because of the relatively low volume of arithmetic processing as compared with the total processing and filing time of the real-time system, it seems worthwhile to use a floating-point representation as the format for test results inside the system. There are several arithmetic functions which will be needed by more than one of the subprograms in the system, which would be grouped together as a common arithmetic module. These functions include conversion of text strings to the internal format of the system; conversion of numbers in internal format back to text strings; addition, subtraction, multiplication, and division of numbers in internal format; and perhaps taking the square root of numbers in internal format.

APPENDIX IV - ALTERNATIVE TERMINAL CONFIGURATIONS

As discussed in Section 6, there are two potential problems with the suggested entry room terminal configuration. Although we do not expect that these potential problems will materialize, we have provided for an alternative configuration. In this configuration, a separate label printer is added for each entry station. The entry terminal used in this case would be the same Microswitch keyboard/NCR thermal printer combination used in the laboratory. The operator types input on the terminal and the computer responds to the terminal. Labels, however, are printed on the label printer. Although the label printer can operate at the same speed as the terminals, the total bit rate of 26 terminals and 8 label printers would require a modification to the less expensive single speed multi-line control unit. Furthermore a separate unit would be required for the LINC-8. Thus, the same multi-speed multi-line controller is used as in the recommended system.

The label printer we recommend is manufactured by Di/an Controls Inc. The printer is Model #7305 modified as shown in the attached quotation. It includes a cutting mechanism and thus accepts a continuous roll of simple and less expensive label material. The roll is 5" in diameter. The backing is 2" wide and the label's surface material is 1.6" wide centered on the backing, leaving an edge for easy removal of the labels. As there is no cross-wise pre-cutting of the label material, there is no need for precise positioning of the paper in the label printer.

The label printer is an uncovered rack mounting unit for which we would expect BSL to provide a suitable enclosure.

The cost of the entire terminal subsystem in this configuration is shown below. We would, of course, recommend purchase of at least one Keyboard and one NCR Printer as spares.

Quantity	Model #	Item	Unit Price	Total Price
1	516-670A	Low-Capacity Multi-Line Controller for 36 lines.	13,100	13,100
1	516-670B	36 Line Termination Units	19,575	19,575
1	516-670C	36 E.I.A. Interfaces	3,510	3,510
1	516-670G	Multi Code Option	395	395
1	516-670H	Additional Speed for LINC-8	120	120
26	64SW1-2	Honeywell Microswitch Keyboards	(Note 1)	27,570
26	EM-T2	NCR Thermal Printers	1,650	42,900
8	7305	Di/an Label Printers	(Note 2)	46,120
			TOTAL	153,290

Note 1: First unit costs $3,945 including one-time engineering costs. Further units cost $945 each.

Note 2: $5090 each plus $5400 one-time engineering charge for interface.

There may also be problems with the NCR Thermal printer. Although this printer is based on a highly successful military design, units of the EM-T2 design are only now beginning to be produced for commercial use. Although we do not expect NCR to encounter serious delivery or reliability problems, we are prepared to substitute a different Keyboard/printer combination if such problems should develop. The recommended alternative is the Kleinschmidt Model 311 (ES-238.0000) combination Keyboard/printer which operates at a speed of 27 characters per second. These units are priced at $4,820.00 each. Thus, if these units replaced 18 NCR/Microswitch combinations in the recommended system, the system cost would be increased by $35,250.00.