9-22-1966

A Computerized Partnership Accounting Model For Use On The IBM 360 With Applications To Teaching Beginning Accounting

Janet Eck

Recommended Citation

Eck, Janet. "A Computerized Partnership Accounting Model For Use On The IBM 360 With Applications To Teaching Beginning Accounting." (1966). https://digitalrepository.unm.edu/anderson_etds/45

This thesis by Janet Eck

A COMPUTERIZED PARTNERSHIP ACCOUNTING MODEL FOR USE ON THE IBM 360 WITH APPLICATIONS TO TEACHING BEGINNING ACCOUNTING

By

Janet Eck

A Thesis
Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Business Administration

The University of New Mexico

1966

This thesis, directed and approved by the candidate's committee, has been accepted by the Graduate Committee of the University of New Mexico in partial fulfillment of the requirements for the degree of

MASTER
OF
Business Administration

Brian E. O'Neil
Dean

Date September 22, 1966

A COMPUTERIZED PARTNERSHIP ACCOUNTING MODEL FOR USE ON THE IBM 360 WITH APPLICATIONS TO TEACHING BEGINNING ACCOUNTING

By

Janet Eck

Thesis committee

Perry T. Mori
Chairman

F. Parker Fowler Jr.

Lloyd Seaton, Jr.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

CHAPTER I

THE THESIS DESIGN

Introduction

Students of business and businessmen no longer require assurance that the digital computer will play a vital role in commercial activity. Simple observation will substantiate the acceptance of electronic data processing by private enterprise. At the university, the necessity of providing an elementary introduction to data processing for business students has become clear.

Unfortunately, education in data processing does not always cross disciplinary boundaries. All too often, a student acquires a knowledge of computers in one course and continues to make tedious manual computations in another. Efforts to integrate data processing and accounting education have, to date, been minimal. It is the joint recognition of the role that computers are playing in an accountant's activities beyond the university and of the desirability of providing an application of data processing in the accounting curriculum that has prompted this thesis.

Objectives

The first objective of this thesis is to prepare a comprehensive accounting problem in a format that will permit students of accounting to use a digital computer to prepare standard accounting statements. A programmed problem of this nature could provide a significant vehicle for the integration of accounting and data processing skills. Also, the program itself will provide a valuable example of the logical processes and techniques that are required to automate an accounting system.

Secondly, the thesis will attempt to evaluate the utility of the programmed problem and make recommendations for continued efforts to integrate accounting and data processing within the university. It is hoped that this effort will provide the groundwork for a new approach to the study of accounting.

Scope of the Thesis

The partnership accounting system developed will represent a hypothetical firm. A sufficient number of accounts will be included in the system to permit the demonstration of many of the accounting transactions to which the student has traditionally been introduced in his first semester of college accounting at The University of New Mexico. While it is desirable that the accounting system that is developed in this thesis be believable to the student--that is, the names of accounts and types of transactions that are used should follow accepted accounting practice--it is by no means necessary that the system represent any ongoing firm extant in the real world.

A computer program will be prepared in the Fortran IV, subset E language for the IBM 360, 16K disc operating system. The program will process data relevant to the system and will output the following accounting statements: (1) a general ledger, (2) an income statement, (3) a balance sheet, (4) a statement of accounts receivable, and (5) individual accounts receivable statements. No attempt to perform a ratio analysis upon the resultant data will be made.

The thesis will be limited to a demonstration of a rapid, easy, and inexpensive technique that could be used by students for the preparation of transaction data.

Previous Work Relevant to the Topic

Kemeny, Schleifer, Snell, and Thompson[1] illustrate the use of an accounting matrix similar to the one discussed in the thesis. They also show several simplified flow charts for use in preparing financial statements. All of their remarks refer to a computerized accounting system.

Corcoran, in The Journal of Accountancy,[2] has advocated use of a similar accounting matrix for manual bookkeeping. He feels that through the introduction of matrix bookkeeping to accounting students, the teaching of the mathematics of matrices, linear programming, and operations research will be less difficult.

Mattessich, in Accounting and Analytical Methods,[3] outlines a history of the use of matrices for accounting purposes.

---

[1] John G. Kemeny, Arthur Schleifer, Jr., J. Laurie Snell, and Gerald L. Thompson, Finite Mathematics with Business Applications, (Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1962), pp. 346-362.

[2] A. Wayne Corcoran, "Matrix Bookkeeping," The Journal of Accountancy, Vol. 117, No. 3 (March, 1964), pp. 60-66.

[3] Richard Mattessich, Accounting and Analytical Methods, (Homewood, Illinois: Richard D. Irwin, Inc., 1964), pp. 88-96.

The 1964 American Accounting Association Committee on Courses and Curricula recommends the following regarding incorporating computer technology in the accounting curriculum.[4]

> 1. At the undergraduate level, accounting students should be exposed to electronic data processing in three stages:
>
> (a) Concurrent with or prior to the introductory accounting course, students should receive instruction in a basic programming language, and should be able to write simple computer problems in the language.
>
> (b) Accounting instruction in a variety of subject-matter courses should incorporate some computer-oriented problems, where subject matter is conducive to formulation; but such instruction should convey the principles basic to an understanding of the profounder, nontechnical issues to which computer solutions are being applied.
>
> (c) The traditional accounting systems course should continue to include a coverage of electronic data processing as one of the aids to accounting.

In part (b) the Committee warns that the emphasis should be upon the accounting principles applicable to the solution of a problem rather than upon the mechanics of programming the solution.

Nielsen lists several advantages to be gained from integrating computer programming in the elementary accounting course.[5]

> 1. Frequently students get a fallacious impression from problem assignments that accounting is antiquated in its techniques. Employing computer programming to manipulate data displays the dynamic currency of accounting methodology.
>
> 2. Applying the computer to the solving of accounting problems vividly portrays that the medium of information manipulation and storage does not affect accounting concepts.

---

[4] 1964 American Accounting Association Committee on Courses and Curricula, "Electronic Data Processing in Accounting Education," The Accounting Review, Vol. XL, No. 2 (April, 1965), p. 422.

[5] Gordon L. Nielsen, "The Computer in Accounting Education," The Accounting Review, Vol. XL, No. 4 (October, 1965), pp. 872-873.

3. The student is taught concepts without his associating them with particular data. A complete problem, such as accumulating sales and accounts receivable data, is walked through by way of flow-charting and programming steps, thus helping the student to focus his attention on the use of the data rather than the data itself.

4. The instruction creates a new spark of interest. Programming an electronic computer is likely to be more absorbing than balancing a special journal.

At this time, one publisher has prepared a computerized practice set.[6] In this practice set students keypunch instructions for processing certain types of accounting transactions. With the aid of a tape-resident master program and data cards supplied by the publisher, financial statements can be obtained from the computer. The master program prepares the computer for acceptance of the students' program.

Unfortunately, the computer program is not documented with flow charts that would enable the student to trace the logical operations that attend the automation of an accounting system. Also, the practice set is programmed in the SOPAT language (similar to the Symbolic Programming System language) which is not a common programming language. This feature prevents all but those few who are trained in the SOPAT language from reading and understanding the sample program. Only the most simple operations are demonstrated for the student to enable him to write the programs required in the practice set.

It would be more beneficial to the student if he learned a common programming language such as Fortran or Cobal which could be

---

[6] Daniel Teichroew, Jay M. Smith, and Earl Snell, Computerized Practice Set: SMAC Corporation, (New York: McGraw-Hill Book Company, 1964).

used in other areas. Learning an uncommon programming language such as SOPAT limits the use of programming techniques to the aforementioned practice set exclusively.

The computerized practice set requires an IBM 1401 computer with 8,000 positions of storage, a card reader, a printer, and two tape drives as hardware.

The practice set is intended to focus the student's attention on how business events affect the accounts of the firm and to introduce him to the data processing capabilities of computers. The opportunity to do programming is the motivational factor.

Definition of Terms Used

An accounting period is the time required for the completion of one accounting cycle. An accounting cycle incorporates the elapsed time between the preparation of accounting statements. Obviously, the more frequently accounting statements are prepared, the shorter is the accounting period.

A matrix is a combination of vectors. In this thesis we will be concerned with a two-dimensional matrix where one dimension represents account debits and the second dimension represents account credits. The intersection of a row and a column is called a cell.

A matrix update corresponds to the accounting procedure of journalizing and posting of entries in an accounting system.

A model is a representation of something in a form whose attributes are not the same as the attributes of the thing modeled. An accounting system is a model of the transactions of a firm. A model portrays those characteristics of an entity that the model builder deems significant. To this end, an accounting model need not represent all of the activities and relationships of a firm, but, most frequently, is limited to the transactions that have a distinct bearing upon the information required by the desired accounting statements.

A program is defined as a set of instructions for a computer which will cause it to complete a sequence of operations. Within certain limitations imposed by the size and nature of a computer, programs can be prepared in a host of programming languages. The program used in this thesis will be written in the Fortran programming language.

A vector is a one-dimensional, ordered array of numbers. A single row (or column) of a matrix is a vector.

Systems in general. The term "system" is one of the most powerful and most frequently misused terms in the vocabulary of the contrmporary businessman. Because this thesis is fundamentally concerned with systems in general and a specific accounting system in particular, it is deemed prudent to, first, develop a careful general definition of a system and, second, relate the concept of an accounting system to the definition of a general system. Finally, the specific system (a partnership accounting system) will be described.

A system may be described by reference to three attributes. The first attribute is the set of all operands in the system. Operands have the common property of being elements in the system that are acted upon by some factor or factors in the system. The factors--or more properly, operators--that act upon operands constitute the second attribute of a system. The third attribute is the set of transforms. Transforms have the common property of being the result of the transition of an operand in response to the affect of an operator. Every system contains operands, operators, and transforms. Feedback systems also contain a feedback set.[7]

The state or condition of the total system can be described, at a given point in time, by describing the condition of each operand in the system. When any operand in the system is operated upon such that one or more operands undergo a transition to a different condition (i.e., become transforms), the system is said to have experienced a change of state.

Accounting systems in particular. Accounting systems, which comprise a subset in the universe of all systems, can be defined in terms of sets of operands, operators, and transforms. The elements in an accounting system's set of operands have the common property of being transaction data that are operated upon by the system's operators. The operators in an accounting system include debits, credits, balances, adjustments, and closing procedures. The transforms are the financial statements that result when operands (transaction data) are operated upon by the operators.

Before examining the role of whatever feedback elements might be present in an accounting system, it is illuminating to more closely

---

[7] Feedback systems are discussed in detail by Stanford L. Optner, Systems Analysis for Business and Industrial Problem Solving, (Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1965).

inspect the sets of operands, operators, and transforms. The set of operands in the accounting system of a firm that is engaged in a multitude of diverse transactions might contain thousands of elements. Regardless of the number of elements in the set of operands, the set of operators typically contains only a few operators. Further, the number of ways in which operators are allowed to operate upon transaction data is fully specified and limited by the rules that are embodied in accounting principles. To that end, only a limited number of operations are permissible in an accounting system. An accounting system is thus constrained by accounting rules. The rigid constraints of an accounting system allow the accountant to predict the transforms or output of the system with perfect certainty. An accounting system, then, is wholly deterministic.

Simple observation will support the assertion that most accounting systems do not have a feedback set. That is, there is no element that compares the system's output with some criterion and attempts to change the operands set in response to a difference between the transform set and the criterion. Accounting systems that are designed with the intent of "accounting objectivity" or "historical reporting" lack this feedback characteristic and are not affected by the system's environment. Such accounting systems are closed systems.

On the other hand, accountants are frequently called upon to design accounting systems for managerial information (as opposed to stockholders' reports or historical reports). In such cases, the accountant can compare the information content of the system's output against management's criteria and can make recommendations for changes in the system when the transform set fails to satisfy the established criteria. The concept of the accountant as a feedback element in an accounting system suggests opportunities for future study.

CHAPTER II

THE ACCOUNTING MODEL

The Partnership Accounting System

This chapter is concerned with the development of a partnership accounting system. With reference to the previous definitions for systems in general and accounting systems in particular, the partnership accounting system will be defined by specifying the operands, the operators, and the transforms that will be included in the system.

The operands or transaction data in the system are limited to those transactions which would typically occur in a small business owned by two partners. To accommodate unique transactions, a special instruction has been built into the system. This feature will be discussed in detail in a later chapter.

The model is that of a retail store. In addition to office personnel, the business employs salesmen and deliverymen. The building is rented, but the store fixtures and office equipment are owned by the partnership. The partners may draw cash or merchandise for their own use.

The transaction data used in this thesis were selected for demonstration purposes. The transactions do not include all of the possible activities of an ongoing firm, although the system is capable of handling any occurrence by expanding the account matrix and providing necessary links.

Each type of transaction (operator) which is demonstrated in this model is given a unique code number. The transactions and corresponding code numbers are listed in Table I. It may be noted that the first two numerals indicate the broad classification of the transaction while the last two numerals indicate the specific transaction within each broad type. Code number 2111 indicates to the computer that all data have been processed, and the processing ceases.

The format of a data card is illustrated in Figure 1. Columns one through four contain the transaction code number. The date is in columns five through ten. The amount of money involved in the transaction is punched in columns eleven through eighteen with the decimal point in column sixteen. If the transaction involves a subsidiary account, its number is punched in columns nineteen and twenty. Columns twenty-one through twenty-four are reserved for the interest rate on notes, with the decimal in column twenty-one. The number of days expired on the note are punched in columns twenty-five through twenty-eight. A decimal must be placed in column twenty-eight to facilitate computations.

The date must be punched on the data cards requesting financial statements or the housekeeping routine so that the date will appear on the financial statements.

The broad outline of the model is illustrated in Figure 2. After the program is compiled, punched cards containing the data are read by the card reader. The data is operated upon by the central processor

TABLE I

TRANSACTION CODES

CODE	TYPE OF TRANSACTION
	CASH RECEIPTS
1111	Received cash on account
1112	Cash sale
1113	Received cash for note receivable
1114	Interest earned
1115	Received cash for purchase return
1116	Sold office fixtures
1117	Sold store fixtures
1118	A invested cash
1119	B invested cash
	CASH PAYMENTS
1211	Paid cash on account
1212	Purchased merchandise for cash
1213	Paid for transportation in
1214	Purchased office supplies for cash
1215	Purchased shipping supplies for cash
1216	Paid miscellaneous selling expense
1217	Paid office expense
1218	Paid shipping and delivery expense
1219	Paid miscellaneous general expense
1221	Paid store rent
1222	Paid cash for sales return
1223	Paid delivery truck rent
1224	Paid sales salaries
1225	Paid delivery salaries
1226	Paid office salaries

TABLE I--Continued

CODE	TYPE OF TRANSACTION
	CASH PAYMENTS
1227	Paid for advertising
1228	Paid for insurance
1229	A withdrew cash
1231	B withdrew cash
1232	Paid quarterly FICA tax liability
1233	Paid withholding liability
1234	Purchased office fixtures for cash
1235	Purchased store fixtures for cash
	SALES ON ACCOUNT
1311	Sold merchandise on account
	ACCOUNTS PAYABLE
1411	Purchased merchandise on account
1412	Transportation in on account
1413	Purchase discount on account
1414	Purchase return on account
1415	Purchased shipping supplies on account
1416	Purchased office supplies on account
1417	Shipping expense on account
1418	Miscellaneous selling expense on account
1419	Office expense on account
1421	Miscellaneous general expense on account
1422	Purchased office fixtures on account
1423	Purchased store fixtures on account
1424	Prepaid advertising on account
1425	Prepaid insurance on account

TABLE I--Continued

CODE	TYPE OF TRANSACTION
	ADJUSTMENTS
1511	Accrue interest earned
1512	Depreciation, office fixtures
1513	Depreciation, store fixtures
1514	Adjust prepaid insurance
1515	Provision for bad debts
1516	Adjust advertising expense
1517	Adjust office supplies
1518	Adjust shipping supplies
1519	Accrue salaries, delivery
1521	Accrue salaries, sales
1522	Accrue salaries, office
1523	Record new inventory
	MISCELLANEOUS TRANSACTIONS
1611	Merchandise removed from inventory for store use
1612	FICA accrual for delivery salaries
1613	FICA accrual for sales salaries
1614	FICA accrual for office salaries
1615	Record payroll tax expense
1616	Withholding liability, delivery salaries
1617	Withholding liability, sales salaries
1618	Withholding liability, office salaries
1619	Sales returns and allowances
1621	Quantity discount given
1622	Received a note in payment of an account receivable
1623	Uncollectible account receivable written off
1624	Any other transaction where I and J are punched on the following card
	CLOSING PROCEDURES
1711	Distribute profit or loss; close drawing accounts

TABLE I--Continued

CODE	TYPE OF TRANSACTION
	FINANCIAL STATEMENTS
1911	General ledger
1912	Income statement
1913	Balance sheet
1914	Schedule of accounts receivable
1915	Individual accounts receivable
	HOUSEKEEPING
1811	Reversing entries, zero matrices, transfer balances from ending balance column to beginning balance column
	END-OF-DATA CARD
2111	Stops all processing

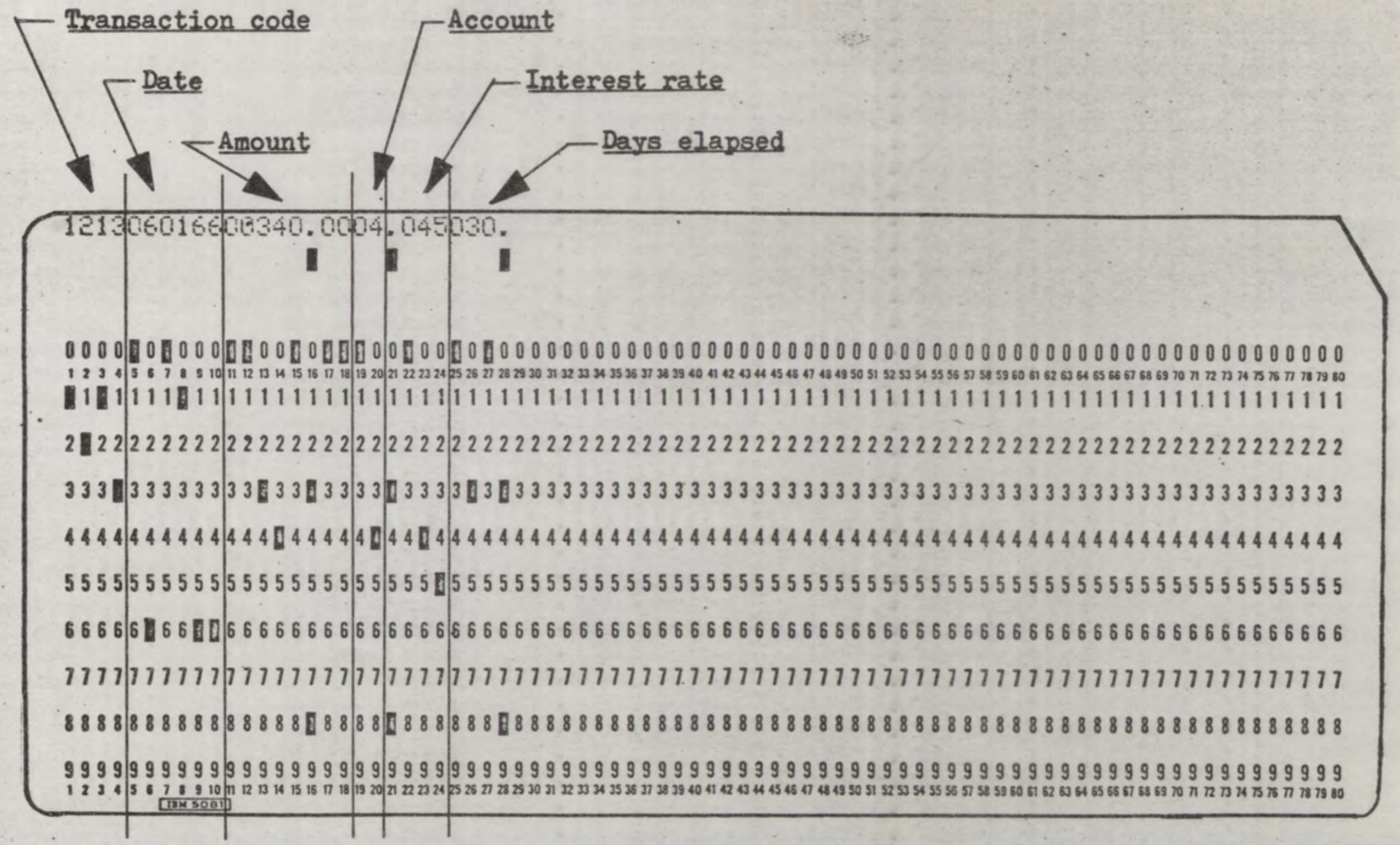

FIGURE 1

DATA CARD

and stored in an internal matrix. Output includes a listing of each card read, financial statements, and [illegible] statements.

The Accounting Matrix

The general ledger accounts of a company are stored in a two-dimensional matrix. A condensed version of the matrix is presented in Exhibit 2. The matrix is referred to as [illegible] matrix in this study.

The rows are debits and the columns are credits. For every transaction, the intersection of the applicable debit and credit accounts is credited with the amount. The amount of the transaction is added to that cell in [illegible] position. It is readily apparent that a single entry contains both the debit and credit portions of the transaction. Therefore, only one entry is required, and entries must be entered on [illegible] transaction basis.

The computer program is designed to select the proper debit and credit accounts according to the transaction code. It is, therefore, not necessary to indicate on the input card which accounts are to be debited and credited.

By making a single entry for each transaction and representing debit and credit portions simultaneously, the [illegible] balance in all [illegible]. This rule of double-entry bookkeeping is demonstrated mathematically by Kemeny, Schleifer, Snell, and Thompson.[8]

---

[8] Kemeny, Schleifer, Snell, and Thompson, op. cit., pp. [illegible].

Beginning balances of creditor firms are always entered in the first few rows of the firm matrix. As there are ten creditor firms which were extended credit in this model, the first ten rows of the matrix contain balances at the beginning of the period. During the first period, these balances are read from punched cards. Thereafter, the balances are transferred to the matrix as part of the housekeeping routine.

Each creditor firm is assigned a unique identification number. The numbers and corresponding firm names are listed in Table III. Whenever a transaction is recorded in the accounts receivable records, the firm number is entered in column one. At the end of the period, the entries are sorted by this number for the purpose of printing statements.

The month, day, and year of a subsidiary transaction are entered in columns two, three, and four. Debits are entered in column five and credits in column six. It is impractical and unnecessary to maintain running balances for each firm in this type of matrix. Consequently, balances are calculated only when statements are printed. If a listing should be required at an interim date, a statement can be called for and printed in a matter of seconds.

Although all credit purchases made by the corporation were from a single supplier, another matrix for accounts payable could be used if the number of suppliers increased. Except for the name, the new matrix would be identical to the firm matrix.

[illegible]

Initial Routine

The initial routine is diagrammed in Figure 5. The logic in this portion of this routine will be [illegible] of the program. Its primary purpose is to [illegible] in all general and [illegible] routines.

The first step is to read a data card containing values for N, [illegible] and KC. N is the number of credit balances to be [illegible], M is the number of credit balances to be read, and KC is the number of accounts receivable subsidiary ledger accounts. These values will be used in later processing.

Next, each cell in the accounts [illegible] is set equal to zero. This is done to assure that no extraneous data enters the processing.

The program then enters a DO loop where N cards are read. Each card contains a [illegible] indicating the [illegible] to the proper account and a value of the credit balance to be [illegible], the balance is stored in the [illegible] specified row number. [illegible] on the printer [illegible].

After N cards have been read and processed, [illegible] is entered where M cards are read. The [illegible] as that of the [illegible] number indicates [illegible] balance, [illegible]

time, is stored at the intersection of [illegible] and the action. As before, the data are printed.

The last step of the [illegible] is to read cards containing balances for each segment [illegible]. The cards contain the identification number of the firm, the current date, and the daily balance. It is assumed in this model that all [illegible] balances are deficit. This information is recorded in the files [illegible] as discussed in the section describing the matrix. [illegible] read and [illegible] to the number of [illegible] firms, balances must be read for every firm even though the [illegible] may be zero.

The variables [illegible], N, and NT were used to make the program more flexible. The number of balances of each type may be varied simply by changing the initial [illegible]. If they were constants, the [illegible] program would have to be changed to accommodate varying numbers of balances.

The final step of this routine is executed each time a transaction card is read. [illegible] card date is read, the transaction code is checked for invalid codes. No zeroes are allowed in the transaction codes in this model. If an invalid code is found, [illegible] an error routine (to be discussed later) [illegible] and the next card is read.

If no invalid codes are found, the program branches to a [illegible] subroutine as indicated by the first two digits of the transaction code. This transfer is managed through the use of a computed GO TO. If an [illegible] error routine. When the first digit of the code is a two, the program goes to a [illegible], and the [illegible].

Cash Receipts Procedure

If the first digit is a [illegible] in [illegible] the program branches to the cash receipts routine that is illustrated in Figure 6. [illegible] is given the [illegible]. [illegible] to be debited, cash in this case. [illegible] of the [illegible]. [illegible] value should appear, the program branches to the error routine.

Finally the value of the last code figure is [illegible] computed by [illegible]. The digits [illegible] routine [illegible] the value of the [illegible] column to be credited, [illegible] to the matrix [illegible].

When the final figure of the code has [illegible] transaction is [illegible] matrix. The [illegible] entered in column [illegible] amount in column [illegible] to ensure that [illegible] the following row.

Cash Payments Routine

When the value of the first two digits of the transaction code is twelve, the program branches to the cash payments routine (Figure 7). The value of J is set at 1 and [illegible] the transaction [illegible] in this routine.

The third figure of the transaction code may be either zero, one, two, or three. Any other value will cause the program to branch to the error routine. The value [illegible] the code may assume values from [illegible]. The fourth figure may assume values from one to three.

A value in excess of [illegible] figure of [illegible] causes the program to branch to the error routine.

Sales on Account Routine

The transaction code for this routine [illegible] sales routine, there is no distinction [illegible] will be the only acceptable code for this routine. Any other figures appearing in the code will cause branching to the error routine.

After verifying that the code is correct, the program sets values for I and J. The data are entered in the [illegible] before. The program then branches to the [illegible] routine.

Accounts Payable Routine

The third figure of the transaction code may assume the value of one or two in the accounts payable routine. A value of [illegible]. For the value of one the fourth figure of the code [illegible]

from one to nine) for the value and if any [illegible] from one to five. As before, an unsatisfactory number causes the program to branch to the error routine.

Values are assigned to X and Y for each transaction, and the program branches to the credit routine.

Adjustment Routine

Transaction codes beginning with the number six [illegible] that nothing will be [illegible] remaining, [illegible] have the first [illegible] the [illegible].

A computed GO TO then checks the value of the fourth figure of the code. Receivable notes are [illegible]. If the value is one, the fourth figure [illegible]. If the value is two, the fourth figure [illegible].

The amounts of [illegible] to be [illegible] be [illegible] notes receivable is [illegible] by the [illegible] code in general [illegible] of the [illegible] interest and [illegible] number of days [illegible] the proper [illegible] and the accrued interest [illegible].

The amount to be [illegible] cards is as follows:

Depreciation of [illegible] depreciation.

Insurance--the amount to be transferred from prepaid insurance to insurance expense.

Bad debts--the amount to be added to the allowance for bad debts.

Advertising--the amount prepaid expired.

Supplies inventories--the amount of supplies on hand.

Accrued payrolls--the amount of the accrual.

Merchandise inventory--the amount of the merchandise on hand.

The correct amounts are compared with the records of each department. After each update a new data card is made.

The merchandise inventory adjustment is made by deleting revenue and expense and [illegible] inventory with the [illegible] inventory [illegible]. The ending inventory is then [illegible] revenue and expense.

Miscellaneous Transactions Routine

Transactions which would normally be recorded in a general journal are gathered together in the miscellaneous transactions matrix, Figure 11.

The first step in this routine is to enter the data cards as needed. The transaction data then is used to find the account [illegible] to which the [illegible] occurs. For each transaction the amount is added to the proper cell in the account matrix. If a [illegible] is involved, the [illegible] matrix is updated. [illegible] If the transaction is incorrect, [illegible] with each entry a new data card is made.

The host transmits in this manner until [illegible] this allows a type of [illegible] commands [illegible]. After reading a data card [illegible] correct, a second card is read which contains the value [illegible]. It will be handled until [illegible] of the accounts to be debited and credited, respectively. The [illegible] is then updated.

This process would be used when [illegible] or for corrections entered [illegible]. It was not [illegible] to program every possible type of transaction into the model; [illegible] commands which is possible [illegible] transaction.

Closing Procedures

This [illegible] in printing the financial statements. [illegible] are printed. All row and column [illegible]. The [illegible] stored at the intersection of the [illegible] matrix.

Next, [illegible] to determine whether each account has a debit or a credit balance. Debit balances are stored at the intersection of the row vector of the account and column vector forty-eight. Credit balances are stored at the intersection of the column vector of the account and row vector forty-eight. If debits equal credits, a zero is entered in both places.

When all accounts [illegible] is distributed between the [illegible] ratio. In this model the ratio is 60 per cent [illegible] and 40 per cent [illegible].

To determine the profit or loss, all expenses are added, and then all revenues are added, their [illegible] expense charges, which are subtracted from revenues. If revenues are greater, the difference is profit. If expenses are greater, the difference is loss.

If the difference is profit, the capital account balance is tested to determine whether it is debit or credit. If the balance is a credit, the sum is calculated and moved to a credit balance. When the capital balance is a debit, the amount of the profit is compared with the debit balance. If the profit is greater than the debit, it is subtracted from the debit balance. If the profit is smaller than the debit balance, the difference is debited, or if equal, the debit balance is set to zero.

The same procedure is followed to update the capital account.

A similar procedure is followed when a loss occurs. In this case the loss for each month is subtracted from a credit balance or added to a debit balance.

The final step of this routine is to close the proprietor's drawing account. The balance in each account is tested to determine whether it has a debit or credit balance. If the balance is a debit, the balance of the drawing account is moved to the capital balance. (In this model the drawing account balance [illegible].)

When the capital is credit in amount, the capital and drawing balances are compared. If the capital credit is greater than the drawing balance, drawings are subtracted from the capital credit balance. If drawings are greater than the capital credit, the difference is debited to the capital debit balance, and the capital credit is set to zero.

The same sequence of steps [illegible] for Line B's [illegible] account.

Housekeeping Routine

The housekeeping routine illustrated in Figure [illegible] is the [illegible] portion of the program to be executed [illegible]. It [illegible] the entries for the following [illegible] and performs those adjustments [illegible] need to be recorded.

First, all of the current month's [illegible] column is set to zero. The individual [illegible] then reversed: interest receivable, [illegible], [illegible], shipping expense, and returns payable.

The closing [illegible]. The [illegible] balances are [illegible] corresponding beginning balance [illegible]. The [illegible] and column are then set to zero. [illegible] nominal accounts [illegible] beginning balance column.

The third part [illegible] balance of each [illegible] place is [illegible] interest for future [illegible].

Financial Statements Routine

The financial statements routine, [illegible] to the financial statements [illegible] by the [illegible].

should be noted that its statements may be printed as they are [illegible] that the statement of accounts [illegible] must total the [illegible] individual statements of [illegible].

<u>General Ledger Routine</u>

The general ledger routine (Figure 13) prints the total debits, total credits, beginning balance, and ending balance for each account in the account matrix.

The first command causes the [illegible] column headings to be printed.

Next, the beginning balance of an account is checked to determine whether it is a debit or credit. If it is a credit, the amount is multiplied by minus one. This then identifies the credits with a minus sign. The same check and multiplication is performed on the ending balance.

A more suitable [illegible] account title [illegible] titles or debits are in the same manner as [illegible] the matrix.) The account title, total debits, total credits, beginning balance, and ending balance are printed.

The above procedure is repeated for each account in the account matrix. Running totals are kept for each column. After all the accounts have been printed, the totals are printed. [illegible] totals will be equal according to the [illegible] system set forth in this thesis. The beginning [illegible] should be checked for the equality of debits and credits balance [illegible].

The set of books established [illegible] if the computer used [illegible]

time this thesis was written, the Fortran compiler for the IBM 360 computer which was used did not have this capability. If alphameric characters could be stored, the general ledger could be changed. They could then be edited for use as needed in this financial statement.

Income Statement Routine

The income statement routine, Figure 17, begins by writing the statement heading. Titles and dollar amounts are printed. Since alphameric characters cannot be stored, the account titles are included in the format statements and printed explicitly. Sales returns are subtracted from sales, and the difference is net sales.

Beginning inventory, purchases, freight on purchases and purchases discounts are printed. Purchases returns and purchases discounts are added. This total is subtracted from purchases to obtain net purchases which is printed. Beginning inventory, net purchases, and freight in are added to find the cost of goods available for sale. Ending inventory is subtracted from the goods available to find cost of goods sold. Cost of goods sold is subtracted from net sales and the result is gross profit. Totals are printed as each is computed.

Each selling expense is printed, then the total, and then subtotaled and printed. General expenses are printed, totaled, and the total is printed. The subtotals are added to find total operating expenses. Total expenses are subtracted from gross profit to determine net operating income. Interest is then added to the net income to find net income.

The above figures are added to accumulate working capital balances for A and B. The two balances are summed to obtain equity. Total liabilities and total assets are then added.

Schedule of Accounts Receivable Routine

The schedule of accounts receivable routine, Figure 18, can be executed after the individual statements of account have been printed. The routine begins by printing the necessary headings.

Next, a card is read [illegible] the [illegible] cards are in sequence according to the firm identification number. The firm name and its corresponding balance are printed.

When all of the balances have been printed, they are totaled. Total accounts receivable is then printed.

Individual Accounts Receivable Statements Routine

Figure 19 outlines the procedure for producing individual firm statements of account. A card is read which contains the firm's name. This name is printed as part of the statement heading.

A DO loop is entered which [illegible] the number of firms. For each value of the [illegible] recorded in each row of the firm [illegible]. Each time the numbers are summed, [illegible] is printed on the statement. A running balance is also printed. This procedure results in ten statements of account.

The final balance for each firm is stored in a balance list. The position in the list corresponds with the firm number. These balances are used then printing the schedule of accounts receivable.

The financial statements may be requested at any time during an accounting period. They may be asked for singly or all together. The order of the requests for the statements may be random except for the schedule of accounts receivable. Because this listing is a by-product of the processing of individual customer statements. Therefore, the individual statements must have been printed before the schedule is called for.

Closing procedures must have been executed before the financial statements are printed if meaningful statements are to be prepared. The housekeeping routine may be performed only after all desired financial statements have been printed. The routine destroys data essential to the preparation of the statements.

The end-of-data card, code 2111, should be placed at the end of all other data. The program enters housekeeping mode, and it may not be started again without recompiling the entire computer program.

A transaction code may not contain a zero. When the transaction card is read, the code is checked for zero. If equal to zero, the error routine is executed. The computer operator may then correct the card and resubmit it for processing.

When a transaction code contains a number which the program rejects as being infeasible, the error routine is also executed. Again, the card may be corrected and resubmitted.

If an error is made in a transaction code, but the error is still an acceptable code, the data will be incorrectly processed. There is no way of detecting this type of error before the data is processed.

It is similar to the errors of transposition [illegible] [illegible] [illegible] by a manual method. [illegible] of error may be found by analysis. [illegible] print out of the [illegible] and examining it with the individual document recording the transaction.

The need for a trial balance to catch the type of error which a trial balance is intended to expose, that of inequality of debits and credits, cannot occur in this system.

The print out of each data card provides a record of transactions for audit purposes or future reference. Financial statements may be requested daily if necessary for intelligent day-to-day decisions.

The correspondence of the several programs' routines to the special journals used in an accounting system should be noted. The cash receipts routine contains transactions normally entered in the cash receipts journal; the cash payments routine contains transactions normally recorded in the cash disbursements journal; the sales on account routine corresponds to the sales journal; and the accounts payable routine corresponds to the accounts payable journal. The adjustments and miscellaneous transactions routine contains entries normally entered in the general journal.

By grouping similar transactions under one heading (routine), the task of finding errors and omissions is simplified.

The various documents and the forms used in computations throughout the [illegible] cycle, [illegible] [illegible], [illegible] actual real-world [illegible] [illegible] [illegible]. [illegible] following the logic of the [illegible]. [illegible] [illegible] the program also aids [illegible] [illegible] of the [illegible].

The cards could be marked with each module of the program intended to aid in the accuracy of the marking. The pencil-marked cards would then be passed through the IBM 519 document originating machine (reproducing punch) or a similar machine which would prepare correctly punched cards from the mark-sensed cards.

A demonstration using students' own work, locally, provides the motivation for further work with data processing techniques.

A second step in the integration process, suitable for the second semester of accounting, would be to have the students write simple routines for very limited segments of the accounting cycle. These programs might be for the posting of ledger accounts from the journal, or for payroll computations. The programs that are assigned must necessarily be limited in scope so that the student is able to complete the program within a reasonable amount of time.

In later accounting courses more complex programs may be assigned. Cost accounting problems may be assigned as well as tax problems. In the more complex areas, the program should be written to solve a specific problem. It might be quite lengthy (especially so) programs require an above-average programming skill and available time.

By programming solutions to problems, it is expected that the student will understand more of the processes involved than just the expected answer. This should aid in the transfer of knowledge from the solution of a specific problem to the technique of solving related problems. This is, perhaps, the ultimate goal of education for the pupils.

CHAPTER IV

RESULTS

Introduction

Each data card is printed exactly [illegible] (see Figure [illegible]). The first cards are the [illegible] [illegible] balances entered in the [illegible]. The next group are the [illegible] [illegible] in the subsidiary accounts [illegible].

The [illegible] [illegible] in the [illegible] [illegible] errors were added to [illegible] [illegible] [illegible] [illegible] routine. [illegible] [illegible] [illegible] purposes. [illegible] accounts debited and credited is [illegible].

In [illegible] [illegible] [illegible] [illegible] [illegible] purposes or for future reference. [illegible] transactions [illegible] [illegible] [illegible].

General [illegible]

The general ledger [illegible] the total [illegible] credits, including [illegible] balances, of each [illegible]. [illegible] and ending balances [illegible]. The [illegible] same sequence as the [illegible] in the [illegible].

The total of each [illegible] is [illegible]. [illegible] total credits. The total of the [illegible] [illegible] is zero. This proves the [illegible] debits and credits.

Income Statement

The income statement, Figures 22 and 23, illustrates the use of literals in format statements. Subtotals and totals are again calculated. The dollar sign is placed, as [illegible], at the [illegible] beginning of a column. The computer is not capable of [illegible] the under-scores above totals at this time.

Figure 25 illustrates the more [illegible] profit [illegible]. Figure 26 illustrates a loss. Although not [illegible] as labeled "NET INCOME", a loss is indicated by a minus sign (see Figure [illegible]).

Balance Sheet

The balance sheet is illustrated in Figures 27 and 28. The normal balance sheet format is used. It is also [illegible] the use of literals in format statements. Dollar signs again indicate totals at the beginning of columns.

Figure 27 is the usual format for the balance sheet. The liabilities section of Figure 28 illustrates the [illegible] with a [illegible]. In the case of an [illegible], each [illegible] of the [illegible] is [illegible]. It is [illegible], [illegible] being the [illegible] liability.

In the eighth position, columns are used to [illegible] figures and [illegible] figures to indicate where they [illegible].

Individual Statements of Account

Individual statements of account, Figure 29, are suitable for any company that is extending credit to its customers. Such statements would normally be printed on a separate form. The program [illegible] this thesis is designed to show the effect.

BIBLIOGRAPHY

Books

Ashby, W. Ross. *An Introduction to Cybernetics*. New York: John Wiley & Sons, Inc., 1956.

Kemeny, John G., Arthur Schleifer, Jr., J. Laurie Snell, and Gerald L. Thompson. *Finite Mathematics with Business Applications*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1962.

Mattessich, Richard. *Accounting and Analytical Methods*. Homewood, Illinois: Richard D. Irwin, Inc., 1964.

Optner, Stanford L. *Systems Analysis for Business and Industrial Problem Solving*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1965.

Teichroew, Daniel, [illegible], and [illegible]. *[illegible] Practice [illegible] Computation*. New York: McGraw-Hill Book Company, [illegible].

Periodicals

Corcoran, A. Wayne. "Matrix Bookkeeping," *The Journal of Accountancy*, Vol. 117, No. 3 (March, 1964), pp. 60-66.

Nielsen, [illegible] "The [illegible]," *The Accounting Review*, Vol. [illegible], No. 4 ([illegible]).

1964 American Accounting Association Committee on [illegible] Curricula. "Electronic Data Processing in Accounting Education," *The Accounting Review*, Vol. 41, No. 1 ([illegible]).