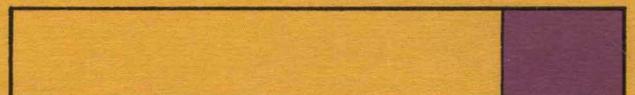


Burroughs

B 251

**VISIBLE RECORD
COMPUTER SYSTEM**

Reference Manual



BULLETIN 251-21001-P
DECEMBER 20, 1960

**THE BURROUGHS B251
VISIBLE RECORD COMPUTER SYSTEM
REFERENCE MANUAL**

*Sales Technical Services
Equipment and Systems Marketing Division*



Burroughs Corporation

DETROIT 32, MICHIGAN

TABLE OF CONTENTS

	Page
SECTION 1 — INTRODUCTION	1-1
B 251 Visible Record Computer System	1-1
Data Processing Systems	1-1
Data Processor	1-2
Program Card Reader	1-2
Sorter-Reader	1-2
Record Processor	1-4
Console and Keyboard	1-4
 SECTION 2 — STORAGE AND INPUT-OUTPUT MEDIA	 2-1
Internal Storage	2-1
Punched Card	2-2
MICR Documents	2-2
Visible Record	2-4
 SECTION 3 — PROCESSING FUNCTIONS	 3-1
Logic	3-2
Arithmetic	3-2
Masking	3-2
Input Control	3-2
Output Control	3-2
Internal Checking	3-2
Programmed Checking	3-2
 SECTION 4 — PROGRAMMING	 4-1
Instruction Format	4-1
Operation Code	4-1
Variants	4-1
Addressing	4-1
Instructions	4-2
No Operation	4-2
Add	4-2
Subtract	4-3
Multiply	4-5
Divide	4-5
Compare-Branch Equal	4-6
Branch	4-7
Transfer	4-8
Mask	4-8
Halt	4-9
Card Read	4-10
Sorter Read, Demand	4-10
Sorter Read, Flow	4-10
Control Sorter	4-11
Record Processor Read	4-11
Record Processor Write and Eject	4-12
Record Processor Print	4-12
Skip Record Processor	4-12
Sample Problem	4-13

TABLE OF CONTENTS (Continued)

	Page
SECTION 5 — SYSTEM COMPONENTS	5-1
B 101 Sorter-Reader	5-1
Sorter-Reader Control Panel	5-1
Data Processor	5-2
Data Processor Console	5-2
Program Card Reader	5-4
Program Card Reader Functional Operations	5-4
Program Card Reader Display Panel	5-4
The Record Processor	5-5
Record Processor Functional Operation	5-6
Console and Desk	5-9
Console Control Panel	5-9
Console Keyboard	5-12
SECTION 6 — ASSEMBLER FOR THE B 251 VISIBLE RECORD COMPUTER SYSTEM	6-1
Introduction	6-1
Assembler Programming Procedures	6-1
Assembler Coding Form	6-1
Control Entries	6-4
Special Addressing Techniques	6-4
Pseudo Op Codes	6-7
Assembly Operations	6-7
B 251 Assembler Instruction List	6-8

APPENDIXES

APPENDIX A

Word Formatting of MICR Information in Storage	A-1
--	-----

APPENDIX B

B 251 Command List	B-1
------------------------------	-----

APPENDIX C

VRC Code Reference Table (Collating Sequence)	C-1
---	-----

APPENDIX D

Flow Chart Standard Symbols	D-1
Input-Output	D-1
Logic	D-1

APPENDIX E

Timing Chart	E-1
Internal Operations	E-1
Input-Output Operations	E-1

LIST OF ILLUSTRATIONS

Figure		Page
1-1	Functions of a Computer System	1-1
1-2	The Five Functions of the VRC System	1-1
1-3	The Data Processor	1-2
1-4	The Program Card Reader	1-2
1-5	The Sorter-Reader	1-3
1-6	The Record Processor	1-3
1-7	The Console and Keyboard	1-4
2-1	Seven-Core Storage Location	2-1
2-2	Binary-Coded Character Representations	2-1
2-3	Punched Card Codes	2-2
2-4	E 13B Type Font Characters	2-2
2-5	Standard Paper Check Format	2-3
2-6	Encoded Check With Transit Number	2-3
2-7	Amount Field	2-4
2-8	Pre-Encoded On-Us Field	2-4
2-9	Post-Encoded On-Us Field	2-4
2-10	Transit Field	2-5
2-11	Auxiliary On-Us Field	2-5
2-12	Addressing with Bar Code	2-5
2-13	Account Record (With Magnetic Stripe and Bar Code)	2-6
3-1	Basic Demand Deposit Accounting	3-1
5-1	Sorter-Reader Item Processing Route	5-1
5-2	Sorter-Reader Control Panel	5-2
5-3	Data Processor Console	5-3
5-4	Program Card Reader	5-4
5-5	Program Card Reader Display Panel	5-5
5-6	The Record Processor	5-6
5-7	Record Processor Schematic	5-7
5-8	Examples of Different Form Sizes	5-8
5-9	Console and Desk	5-9
5-10	Console Control Panel	5-10
5-11	Power and Magazine Switches	5-11
5-12	Console Keyboard	5-11
6-1	Burroughs Assembler Coding Form	6-2



B 251 VISIBLE RECORD COMPUTER SYSTEM

SECTION 1

INTRODUCTION

B 251 VISIBLE RECORD COMPUTER SYSTEM

The B 251 VRC is a completely transistorized, stored-program computer system consisting of the B 250 Data Processor, the B 401 Record Processor and Keyboard Console, the B 101 Sorter-Reader, and the B 122 Program Card Reader. This system is designed to handle, as input media, MICR (Magnetic Ink Character Recognition) encoded documents, punched cards, magnetically encoded account records, and keyboard entries, for supervisory control. As output media, the B 251 VRC System is unique in that it automatically produces a printed hard copy account record with updated, magnetically encoded account information and printed journals for management statistics and reports.

This system provides for the first time a means of combining electronic data processing with MICR and producing hard copy accounting records at high speeds. Another first with the B 251 VRC System is the addition of a magnetic stripe on the back of the account records for storing pertinent identification and account information. This provides a means of communication between the hard copy account record and the automatic equipment. Thus one medium provides storage for automatic processing and visual reference.

This manual is intended to introduce and define the B 251 VRC System to those who desire specific information about it, and to describe in easily understandable terms the complete system and its operation for those who will use it. Prior to reading this manual it would be to the reader's advantage to be familiar with the Burroughs B 101 Sorter-Reader Operator's Manual and the American Bankers Association specifications for the Common Machine Language (Bank Management Publication 147).

DATA PROCESSING SYSTEMS

Data processing systems require five basic functions: input, storage, arithmetic, output, and control (Figure 1-1).

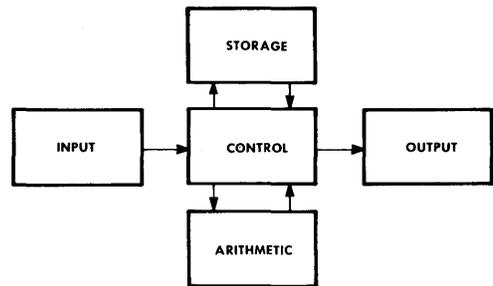


Figure 1-1. Functions of a Computer System

The input function transmits data to the system by means of several input devices, depending on the desired input media.

The storage section holds the data received from the input units and keeps it available for operations by other parts of the system.

The arithmetic function handles the actual processing, or manipulation, of data. It is the computing function of the system which accomplishes all problem solving.

The output function transmits processed results from storage to the output devices and converts them to a usable form.

The control function directs the flow of data from input to storage, storage to arithmetic, arithmetic to storage, and from arithmetic or storage to output.

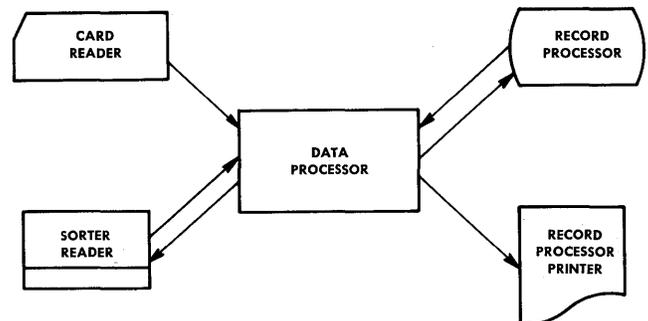


Figure 1-2. The Five Functions of the VRC System

Figure 1-2 illustrates the manner in which these five functions comprise the Burroughs B 251 Visible Record Computer System.

The following are brief descriptions of the component units of the B 251 VRC System, including an explanation of how each performs one or more of the five basic functions of data processing systems.

DATA PROCESSOR

The Data Processor (Figure 1-3) is the nerve center of the B 251 VRC System. Within its transistorized circuitry are logic, checking, and decision-making abilities and the ability to perform the four basic arithmetic functions. This single unit controls all input, output, and formatting. It contains 4800 characters of alphanumeric magnetic core storage. On the front of the Data Processor is a display panel indicating its operating condition at all times and providing control for manual operation.

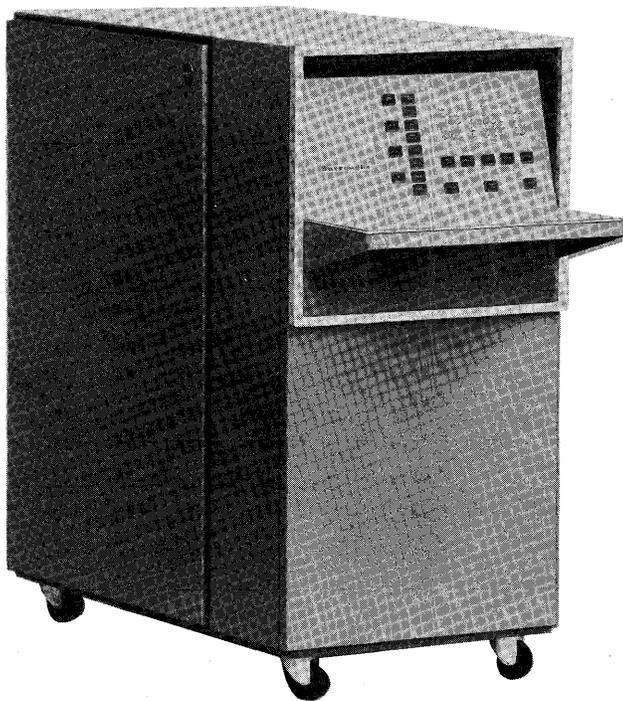


Figure 1-3. The Data Processor

The Data Processor also provides for buffering of the information from the card reader and the Sorter-Reader. That is, information read from a card or an item is held in the Data Processor buffer until it is called for by the required read instruction.

PROGRAM CARD READER

The Program Card Reader (Figure 1-4) is an input device for the B 251 VRC System. It is used primarily for loading programs (the instructions telling the system what operations to perform and in what order) from punched cards. It can also be used for entering exceptions or other extra data into the system. The card reader operates at a speed of 100 cards per minute. The card reader also has a display panel for indicating its operating condition.



Figure 1-4. The Program Card Reader

SORTER-READER

The Sorter-Reader (Figure 1-5) is a high-speed electronic device that reads and sorts magnetically encoded documents at a speed of up to 1560 per minute. While operating in the off-line or sort mode, the Sorter-Reader will sort documents into a pocket corresponding to the value of a selected digit position. In the external mode, the Sorter-Reader acts as an input device for the B 251 VRC Systems, the feeding, reading, and sorting of documents being controlled by program logic stored in the Data Processor. The Sorter-Reader has 13 pockets: 0 through 9, the X and Y pockets for special purposes, and a reject

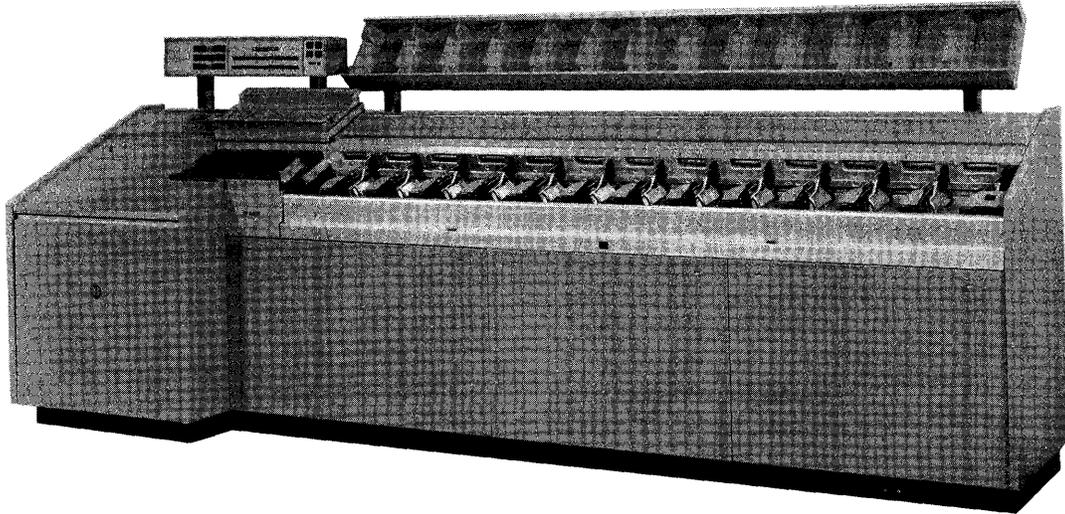


Figure 1-5. The Sorter-Reader

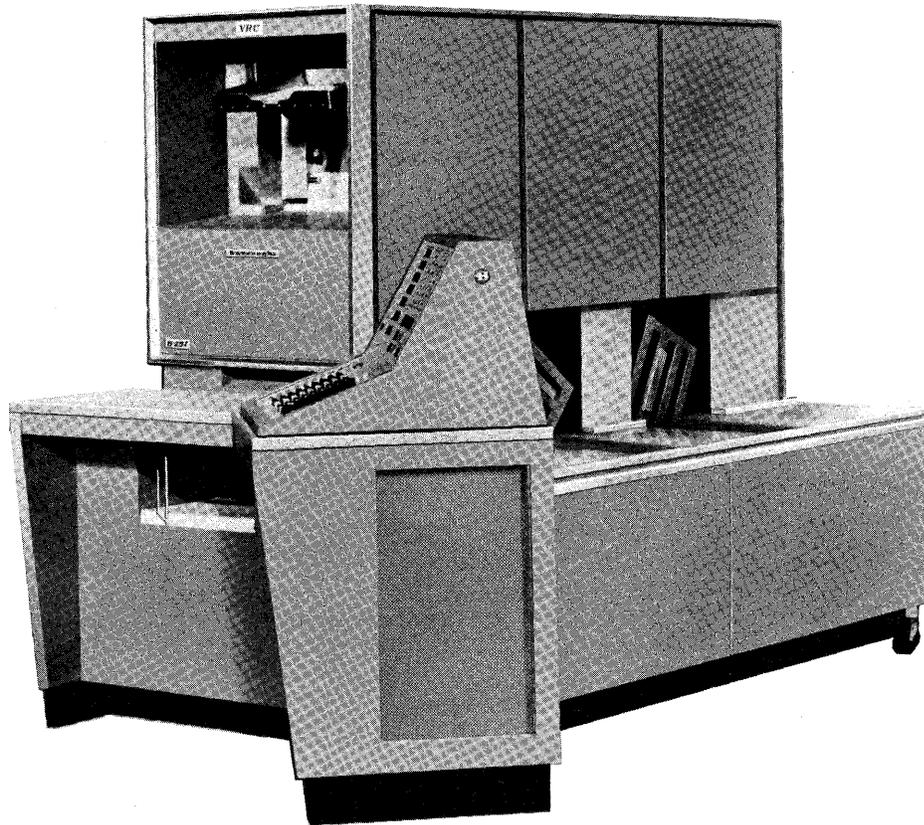


Figure 1-6. The Record Processor

pocket. Above and to the rear of the pockets is a mobile carrier, for ease of handling the documents, and a temporary storage area. The control panel provides indicators and operating buttons for efficient operation of the Sorter-Reader. Field and digit position selection buttons are provided for off-line operation.

RECORD PROCESSOR

The Record Processor (Figure 1-6) automatically processes account records. Records are fed from a magazine in the primary feed station, from the auxiliary feed station, or from the manual feed station. The Record Processor will read account information from the magnetic stripe on the back of the visible record and store it in the core memory of the Data Processor. This unit contains a 160-position printer, which prints on the face of the record and two separately controlled continuous journals simultaneously or individually. Updated account information can be written from core memory to the magnetic stripe and the record ejected to a magazine in the primary stacker station or segregated to the auxiliary stacker station.

CONSOLE AND KEYBOARD

The Console and Keyboard (Figure 1-7) provide the communication between the B 251 VRC System and

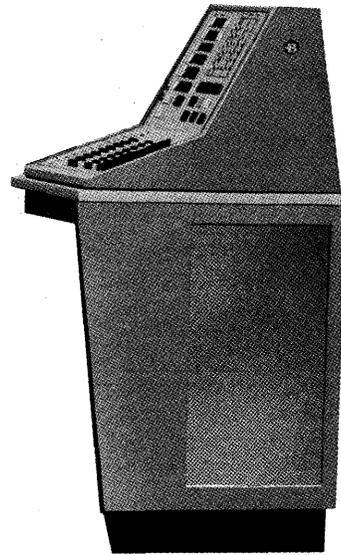


Figure 1-7. The Console and Keyboard

the system supervisor. The Console contains the controls for starting and stopping the system, indicators for displaying the status of the system, and a means for manual intervention or operation of the system. Entry of data into the system is normally through the Sorter-Reader, Program Card Reader, and Record Processor, but the Keyboard permits manual entry, if and when desired.

SECTION 2 STORAGE AND INPUT-OUTPUT MEDIA

INTERNAL STORAGE

The B 251 Visible Record Computer System utilizes magnetic cores for internal storage or memory, each core storing one bit of information. A bit is one-seventh of a binary-coded alphanumeric character.

The binary number system which is built on only two numbers, zero and one, is employed by the B 251 System for representing information within the computer. This is done because of the simplicity of representing the two numbers by any two-state device. The magnetic core is an example of a two-state device. It is used to represent a one (is on) when it is magnetized in one direction, and represents a zero (is off) when it is magnetized in the other direction.

It takes seven binary bits to make up one binary-coded alphanumeric character; thus it takes seven magnetic cores to form one storage location (Figure 2-1). The first four bits are assigned the values 8, 4, 2, 1, respectively, each bit representing that value only when the core is in the one or on state. These first four bits can store any binary-coded decimal digit.

The next two bits are called the zone bits. These bits in combination with the first four can store any binary-coded alphabetic or special character normally used. The seventh bit (parity bit) provides for

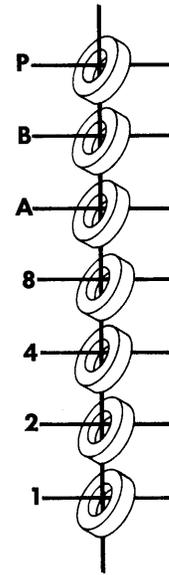


Figure 2-1. Seven-Core Storage Location

internal checking on the bit configuration, being on or off as necessary to make the number of on bits in any configuration always odd. Figure 2-2 shows the binary-coded character representation employed by the B 251 System.

Each character position in magnetic core storage is individually addressable. That is, each storage loca-

CHARACTER BITS												A B C D E F G H I J K L M N O P Q R S T U V W X Y Z																									
		0	1	2	3	4	5	6	7	8	9																										
CHECK	C	1	0	0	1	0	1	1	0	0	1	1	1	0	1	0	0	1	1	0	1	1	0	1	0	0	1	1	0	0	1	0	1	0	0	1	
	B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
ZONE	A	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1		
	8	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	1		
	4	0	0	0	0	1	1	1	1	0	0	0	0	0	1	1	1	0	0	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1	0	0
	2	0	0	1	1	0	0	1	1	0	0	0	1	1	0	0	1	1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
NUMERIC	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	

Figure 2-2. Binary-Coded Character Representations

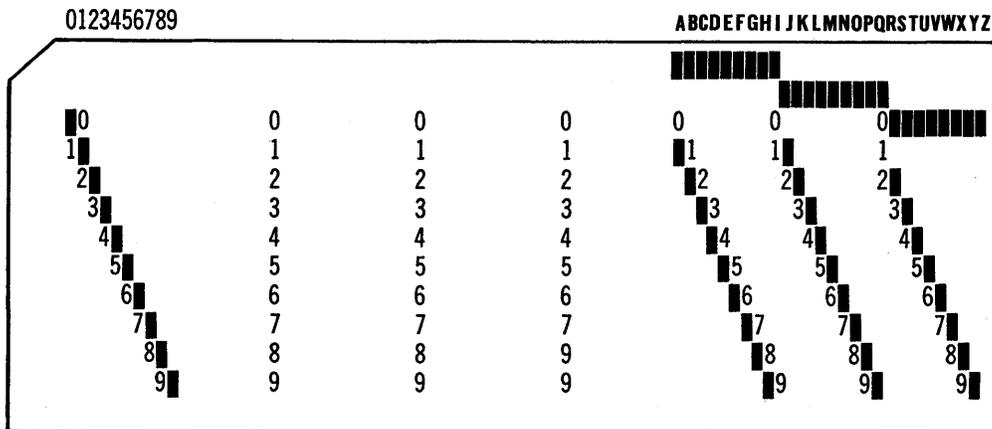


Figure 2-3. Punched Card Codes

tion has an individual label or address by which it might be referenced in order to obtain the information stored there for computation.

A program instruction will designate the precise address of the location storing the information needed for a particular program step.

PUNCHED CARDS

Punched cards can be considered a form of external storage for the VRC System in addition to their primary use as input media. Once a program or set of data has been punched into a deck of cards, that deck can be set aside and used over and over for the same computer application. Alphanumeric characters can be punched into a card for direct entry to computer storage through the Program Card Reader.

The punched card is divided into 80 vertical columns. A total of 80 characters may be punched in one card, each character occupying one of the 80 columns.

Punching takes place in two areas on a card; the lower (numeric) section and the upper (zone) section. The numeric section is divided into 10 horizontal rows, one row for each digit 0 to 9. The zone section is divided into three horizontal rows, 0, 11, and 12. The 0 row is common to both the zone and numeric sections.

See Figure 2-3 for a schematic of card character coding.

Holes punched in their proper rows and specified columns are automatically identified as characters by the B 251 System. Digits are represented by single punches in the numeric section, letters and special characters by combinations of zone and digit punches. The combination of the twelve zone and one of the digits 1 through 9 is recognized by the B 251 System as one of the letters A through I, the eleven zone

with a digit 1 through 9 as J through R and the zero zone with a digit 2 through 9 as S through Z. Special characters are represented by various combinations of zone and digit punches.

MICR DOCUMENTS

The B 251 System utilizes the Burroughs B 101 Sorter-Reader as an input device for reading MICR encoded documents. MICR, upon which the Sorter-Reader is based, represents Magnetic Ink Character Recognition. The Sorter-Reader is designed to read 14 magnetic characters which conform to standards of size, shape, and quality set by the Office Equipment Manufacturers' Committee and the American Bankers Association. These characters consist of the 10 digits 0 through 9 and four special symbols: amount symbol, dash symbol, transit or routing symbol, and the on-us symbol. The 14 characters, printed in type font E 13B, are illustrated in Figure 2-4.

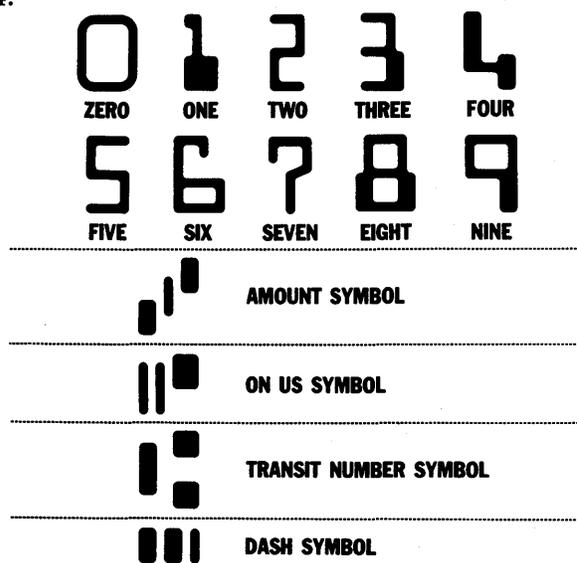


Figure 2-4. E 13B Type Font Characters

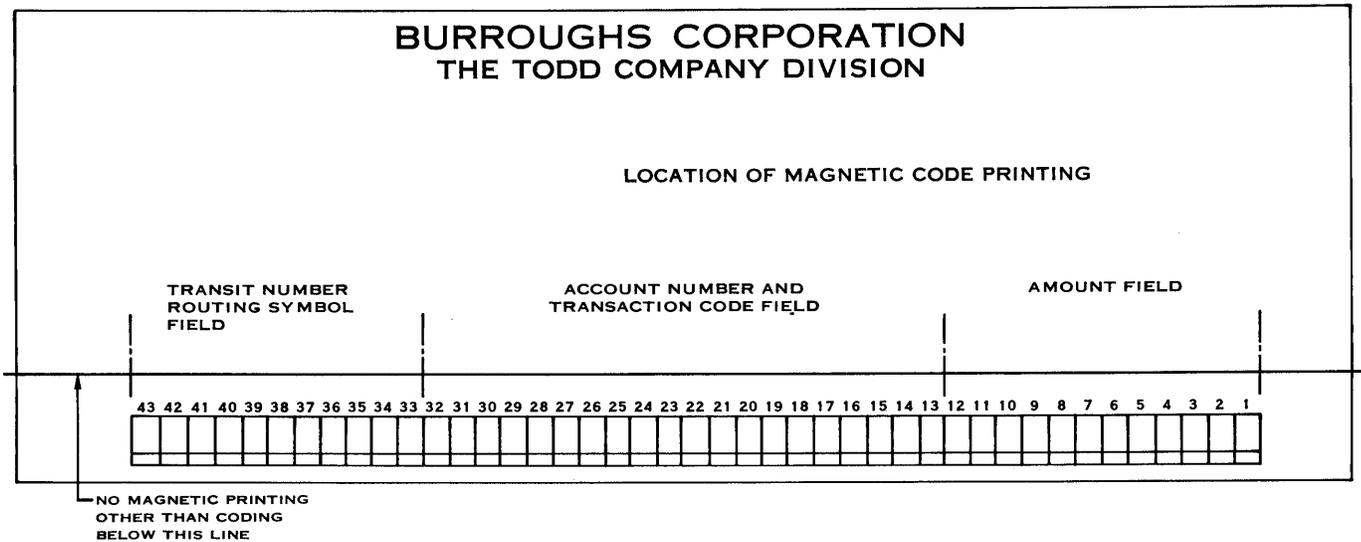


Figure 2-5. Standard Paper Check Format

Figure 2-5 illustrates a standard paper document format used as input media to the B 251 System.

In the following field descriptions it will become obvious that certain encoding positions are ignored. This is due to tolerance requirements between pre-encoded and post-encoded information. For example, a book of checks purchased by a depositor would have the transit and account numbers encoded on each check (Figure 2-6), whereas a counter check would have only the transit number encoded; the account number would have to be encoded after the check had cleared, prior to processing.

Additional information on field location specifications may be obtained from the B 101 Sorter-Reader Manual, Chapter IV, Section 2 and the American Bankers Association Specifications on the Common Machine Language (Bank Management Publication 147).

In the lower right portion of a document is the amount field (Figure 2-7) which occupies positions 1 through 12. Twelve characters must always be present regardless of the amount. Leading zeros are never eliminated. The amount field consists of 10 digits, bracketed by amount symbols.

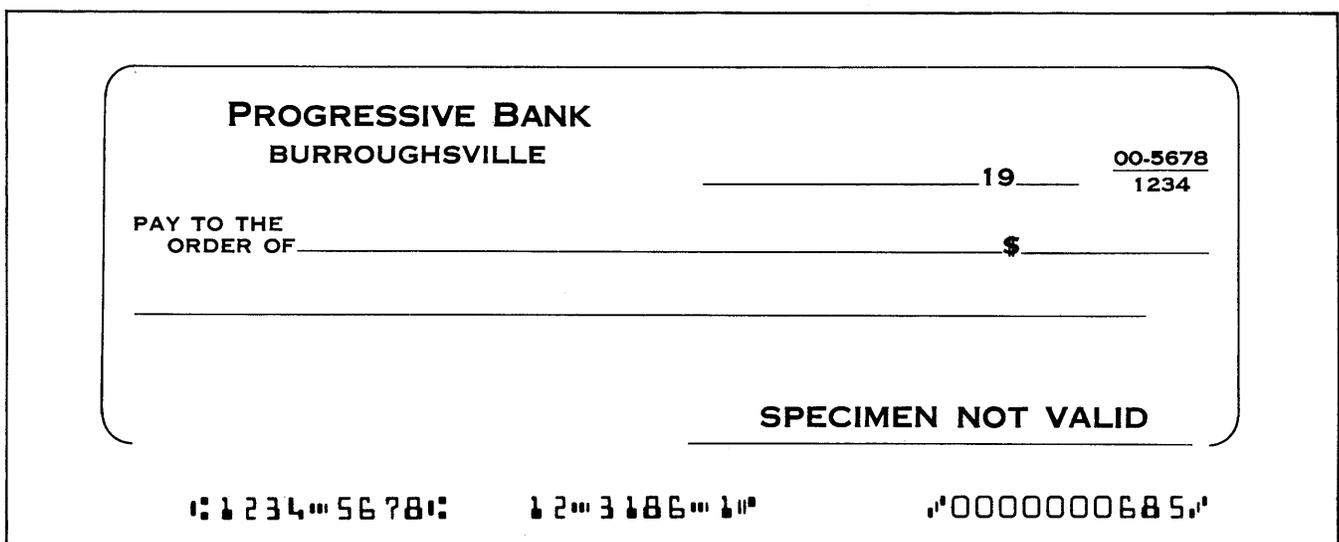


Figure 2-6. Encoded Check With Transit Number

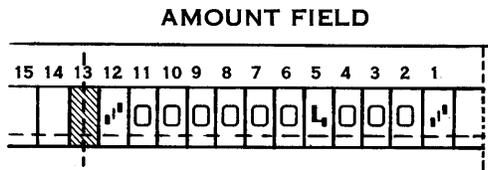


Figure 2-7. Amount Field

The account number or on-us field occupies positions 13 through 32 and includes the transaction code (Figure 2-8). The length of the account number and transaction code may vary but the total number of characters cannot exceed 20, of which only 19 are normally usable due to tolerance requirements. The on-us symbol (position 20 in this example) to the left of the transaction code (positions 14 and 15) is used to define or end the transaction code field. The account or on-us field (positions 21-29) appears to the left of the on-us symbol. The symbol that ends the on-us field and the position it occupies will vary, depending on the length of the account number and when it is encoded. When the on-us and transit fields are encoded at the same time, the transit symbol (position 33) ends the on-us field.

When the two fields are encoded at different times the on-us symbol may also be encoded to the left of the on-us field and to the right of the transit symbol (Figure 2-9).

The transit field (Figure 2-10) occupies positions 33 through 43. Within these 11 positions will be two transit symbols, one each in positions 33 and 43, a dash in position 38, and the eight digits of the transit number in the other positions.

Note that in both the on-us and transit fields, the dash symbol may be used to separate digits to facilitate visual reading.

A fourth field, the auxiliary on-us or account number field (Figure 2-11), is also available on items which

exceed $6\frac{3}{4}$ inches in length. The auxiliary on-us field is located immediately to the left of the transit field and may contain as many characters, not to exceed 16, as can be bracketed between the transit symbol in position 43 and the on-us symbol which is the ending symbol for the field.

The method of storing magnetically encoded document data in core memory is covered in Appendix A.

VISIBLE RECORD

The visible account record (Figure 2-13) is an important form of input-output media employed by the B 251 System. In addition to its primary use as input-output media, it might be considered as external storage for the system in that it stores information in a magnetic stripe which can be read into storage for processing at any time, and it stores visible printed information which can be referred to at any time.

The magnetic stripe (Figure 2-13) is located on the back of the account record. It runs vertically from top to bottom of the record, is approximately .2 inch wide and the center is 2.1 inches from the left edge.

Eighty-one digits of six bits each can be written on the stripe, the bits being packed 60 to the inch. The six bits comprising a digit are the binary 8, 4, 2, and 1 bits, one zone bit and the odd parity bit. The zone bit is used to represent a negative number; absence of the zone bit indicates a positive number. The combination of one, two, three, or four of the 8, 4, 2, 1 bits represents a number 0 through 15. Odd parity, as previously explained, is the assurance that all digits will have an odd number of bits including the parity bit.

Primarily the magnetic stripe is used for storing information pertinent to a specific account. How-

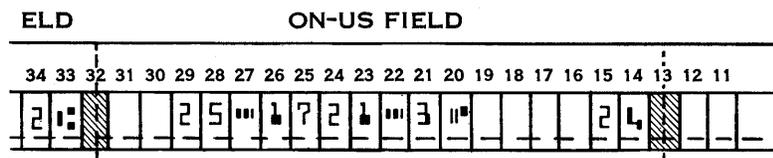


Figure 2-8. Pre-Encoded On-Us Field

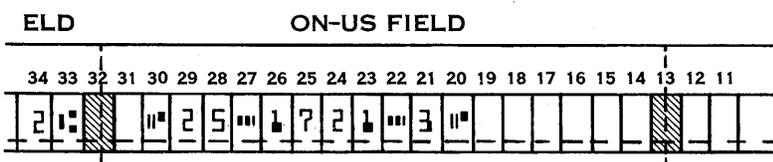


Figure 2-9. Post-Encoded On-Us Field

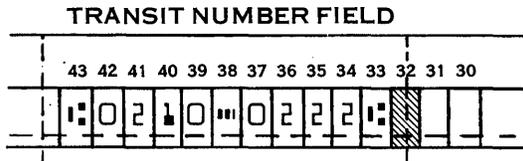


Figure 2-10. Transit Field

ever, it can also be used for other purposes, such as storage of data pertaining to controls, statistics, reports, constants, etc. Information from the magnetic stripe can be read into the Data Processor for processing at any time; and new, updated or partially updated information can be written back on the stripe from storage.

In addition to the magnetic stripe on the back of the visible account record, the record itself is a most important form of output media. All account transactions and balances are printed from Data Processor storage on the face of the record. This is the hard copy produced at high speed which is provided by the B 251 System.

As an optional feature the account record may have a bar code as shown in Figure 2-12. This code is normally the last two characters of the account number. It is used for matching account numbers in core storage with those preprinted on the face of the account record.

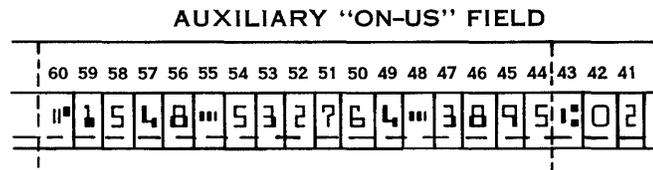


Figure 2-11. Auxiliary On-U's Field

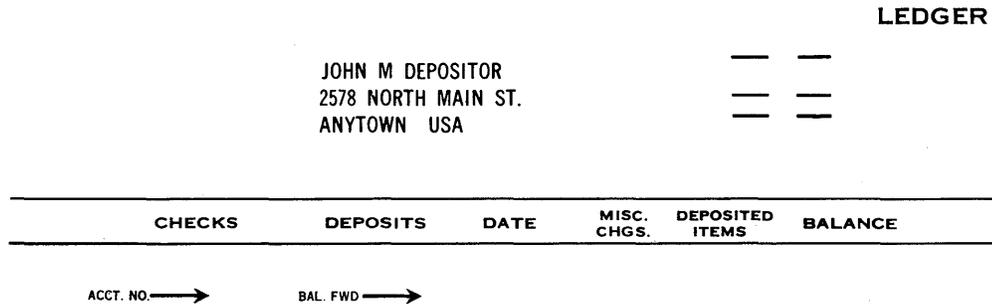


Figure 2-12. Addressing With Bar Code

LEDGER

THE PROGRESSIVE BANK
BURROUGHSVILLE

JOHN M DEPOSITOR
2578 NORTH MAIN ST.
ANYTOWN USA

— —
— —
— —

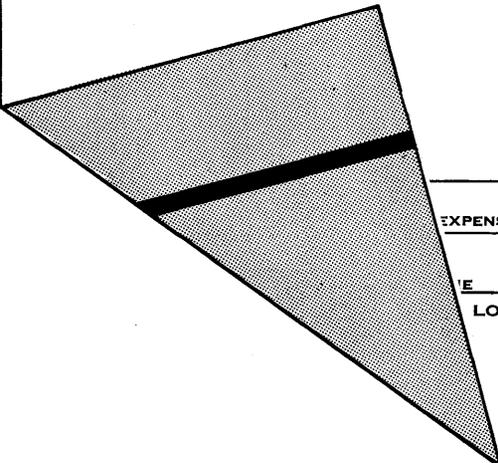
CHECKS DEPOSITS DATE MISC. CHGS. DEPOSITED ITEMS BALANCE CHECKS DEPOSITS DATE NO. OF CHECKS PAID BALANCE

ACCT. NO. →

BAL. FWD. →

BALANCE BROUGHT FORWARD →

Figure 2-13. Account Record (With Magnetic Stripe and Bar Code)



EXPENSE	DEPOSITS	ITEMS	CHECKS
DATE	DATE	PROFIT OR LOSS	BALANCE FORWARD

LOSS CALCULATION

CODE		SERVICE CHARGE	DATE	NO. OF CHECKS PAID
EC - ERROR CORRECTION	OD - OVERDRAWN			
CM - CREDIT MEMO	RT - RETURNED CHECK			
LS - LIST OF CHECKS	SC - SERVICE CHARGE			
CC - CERTIFIED CHECK	DM - DEBIT MEMO			

Burroughs Corporation



DEMONSTRATING FORM

LAST AMOUNT IN THIS COLUMN IS YOUR BALANCE



SECTION 3

PROCESSING FUNCTIONS

Data processing is the performance of a series of actions or operations on data for the purpose of achieving a desired result. The Burroughs B 251 Visible Record Computer System employs the most advanced electronic techniques to perform these functions with the greatest efficiency. In order to control the operations of data processing within the computer system, a series of instructions or commands, called a program, is introduced to the computer to tell it under what circumstances to perform what operations on the data being processed. These instructions will tell the computer to perform operations—such as read, write, print, add, subtract, multiply, divide, transfer, compare, and branch—that are necessary to process the data in the manner desired. Figure 3-1 is a flow diagram of a basic demand deposit accounting program for a bank.

A flow diagram or flow chart is a logical statement of the operations and decisions involved in any one

problem to be solved. Preparation of a flow diagram is the first step in the process of preparing a working program. The analyst who conceives the idea of a computer application sets his plan down on paper in the form of a flow diagram. The flow diagram is then used as a guide in preparing the list of instructions which will tell the computer how to solve the problem. This programming, or coding, is much simpler when a good flow diagram is available: the programmer need only formulate the instructions necessary to accomplish the job set forth in each box of the diagram in the order that the boxes appear. (Figure 3-1). There are several flow diagram symbols which have become standard. (Appendix D).

In the paragraphs that follow, the processing functions of the B 251 VRC System will be discussed. They will be explained in greater detail in later sections of this manual that deal directly with the instructions.

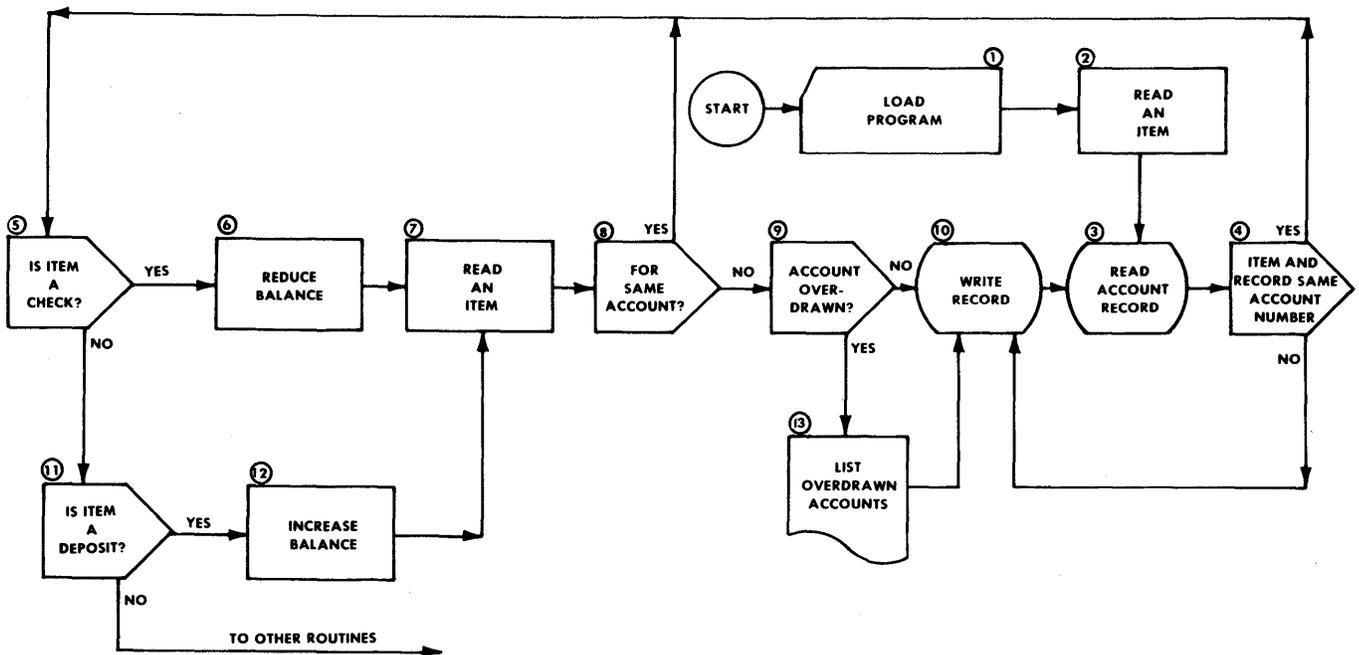


Figure 3-1. Basic Demand Deposit Accounting

LOGIC

Primarily, the function of the logic of a data processing system is to execute program instructions in order. In addition, its purpose is to evaluate and weigh conditions and, if necessary, select alternate routines. Figure 3-1 illustrates typical logical decisions which must be made. For example, step 5 asks, "Is this item a check?" If the answer is yes, the logical sequence continues and the item is processed as a check. If the answer is no, program control will branch to step 11 to determine if the item is a deposit.

The B 251 VRC System's logic is designed to test, prove, and compare, in order to decide if program control should branch or continue in sequence to perform all of the program steps necessary to complete a routine.

ARITHMETIC

The B 251 System has the ability to add, subtract, multiply, and divide. All arithmetic functions are standard equipment; no subroutines are required for multiply or divide. To illustrate two of the arithmetic functions of the system, refer to Figure 3-1, steps 6 and 12. To reduce the account balance, the item amount is subtracted from the balance. To increase the balance, the item amount is added to the balance. The arithmetic and logic units do not store information, but merely operate on the characters as they pass through the computer. Their functions are to receive data from storage, operate on it in accordance with instructions received, and deliver it either back to storage or to an output media.

MASKING

The complete editing (masking) of all output data is under program control. Step 13 of Figure 3-1 includes the masking of the account number and overdrawn balance to insert the desired punctuation and symbols and deliver the masked data to the printer output area in storage for printing. A more detailed explanation of masking will be found in Section 4 of this manual.

INPUT CONTROL

As previously mentioned, the B 251 Visible Record Computer System has great flexibility in that it utilizes the Program Card Reader, Sorter-Reader, Record Processor, and Console Keyboard for a wide variety of inputs. Figure 3-1 illustrates three of these input devices. Step 1 shows how the Program Card Reader is used to load the program into core storage. It can

also be used to read exceptions, daily constants, or other data for an individual application. The use of the Sorter-Reader as input to read MICR encoded documents, such as bank items, accounts receivable, inventory control, etc., is illustrated in steps 2 and 7. In step 3 the Record Processor is used to read account information from the magnetic stripe on the back of the account record. Although it is not used in this example, the Console Keyboard is a valuable input device for manually entering input data.

OUTPUT CONTROL

The B 251 VRC System makes possible a new concept in output media, a hard copy visible record. The B 251 System will render a complete record for normal distribution without the rewriting and editing on auxiliary equipment necessary with other computers. Refer once again to Figure 3-1 for illustrations of output from the Visible Record Computer. Step 10 shows where the updated account information is recorded on the magnetic stripe on the back of the account record. Listing reports and statistics on the continuous journals can be accomplished, as shown in step 13. Although not shown on the flow diagram, the Sorter-Reader is also a form of output control via pocket select.

INTERNAL CHECKING

Parity checks are standard features of the B 251 System. Parity checking is usually construed to mean a form of redundancy checking, based on an odd or even number of binary ones which form a character. For example, in the B 251 System binary representation of a character, a parity bit is made either zero or one, whichever is required to make the number of ones in the character an odd number. Refer to Figure 2-2, which shows the bit representation of a character.

PROGRAMMED CHECKING

In addition to internal checking, the B 251 VRC System provides logic for complete program checking. Branching to alternate routines is included in the Read, Write, and Print commands for end of file, read errors, write errors, and print errors. Through programming, complete flexibility for handling error conditions is available in error routines.

Validity checking is another form of programmed checking. It is a technique based on known reasonable limits of data or computed results. For example: a week has seven days, not eight. A man cannot work 500 hours in one week. There is no day 32 in a month.

SECTION 4

PROGRAMMING

The processing functions of the B 251 System are initiated by program instructions. There are instructions for input and output control, for performing logical operations and arithmetic, and for general control of the complete data processing operation.

INSTRUCTION FORMAT

B 251 VRC System instructions are always 12 characters in length. (See Appendix B for a complete instruction list.) For convenience an instruction is represented as O M N AAA BBB CCC, where:

- O—operation code
- M—first variant
- N—second variant
- AAA—first address
- BBB—second address
- CCC—third address

OPERATION CODE

The operation code is the alphanumeric equivalent of the operation to be performed. The computer recognizes the code character to mean the operation it specifies. For example, 2 is the operation code for the subtract instruction. The operation code is always stored in the most significant digit position of a 12-character field; no instruction will overlap a storage field.

VARIANTS

The second and third character positions, the variant positions, are used to define any variation of the instruction specified by the operation code. For example, 6 is the operation code for the Branch instruction, which is an unconditional branch when M is one and a conditional branch when M is zero.

The variant positions are also used for defining a field length. (A field is a group of adjacent characters and field length is the number of characters in a field.) An instruction referring to a field will address the most significant or leftmost character of the field and employ a variant position to specify the number

of succeeding characters to be included. In an arithmetic operation, the sign of an operand is determined by the zone bit over the least significant digit position of the field: the B bit is on for minus, no zone bit is on for plus.

ADDRESSING

An address refers to the location of a specific character in core storage. The addressing system used by the B 251 System is known as 40-10-12. This means that the total of 4800 character locations in core storage are in effect divided into 40 sections of 10 fields, consisting of 12 characters each. Thus we have a three-character address, AAA, BBB, or CCC, the first character of the address designates the section (Appendix C), the second the field in that section, and the third the character of that field.

When programming, a conversion is necessary in order to obtain the three-character equivalent (AAA) of the four-digit decimal location address (0000 through 4799). This is done by dividing the decimal address by 12. For example, a programmer wants to refer to a character in storage location 3280. Dividing 3280 by 12 yields 273 with four remaining. This means that the address desired is in the 27th section, so that P (see Appendix C) is the first character of the three-digit address. The division also shows that the location is in the third field of the 27th section and is character position four of that field, thus completing the address P34.

Note that in counting the sections, fields, and characters, the first is always zero, succeeding ones being specific numbers, letters, and characters. For example, the last two characters of a field are represented by # and @ respectively.

0	1	2	3	4	5	6	7	8	9	#	@
---	---	---	---	---	---	---	---	---	---	---	---

All examples of instructions used in the instruction descriptions will have decimal addresses for simplicity in understanding.

As indicated by the three addresses represented in the instruction format, the B 251 VRC System is a three-address machine. That is, any one instruction is capable of referencing up to three storage locations. For example, the Add instruction calls for the addition of the contents of location AAA and BBB, and stores the result in location CCC. Whenever M or N are used to designate a field length, AAA, BBB, or CCC designate the most significant digit of the field.

Not all instructions utilize all 12 character positions. Whenever an address or variant portion of an instruction is not used, that portion is irrelevant and may be used for storing extra data, such as constants. For example, the No Operation instruction does not use any of its address or variant positions. Therefore, there are 11 character positions available for storing extra data.

INSTRUCTIONS

Operation Name: No Operation

Mnemonic Code: NOP

Operation Code: blank

Instruction Format:

O M N AAA BBB CCC

O: Operation code blank

M: Not used

N: Not used

AAA: Not used

BBB: Not used

CCC: Not used

Perform no operation. Program control continues to the next instruction in sequence. This instruction is generally used as a switch where an address may be inserted in AAA and, under a certain set of circumstances which dictate that an alternate routine be used, the operation code 9 for a Branch is inserted in the operation code position. The comparison toggles (an internal part of the Data Processor checked by the Compare-Branch instruction) are not affected by this instruction.

Example:

OP M N AAA BBB CCC (No operation is performed)
NOP

When used as a switch it would look like this:

OP M N AAA BBB CCC (No operation is performed)
NOP 1 XXX

When switch is set to Branch it looks like this:

OP M N AAA BBB CCC (The operation code for the Branch Unconditional instruction is inserted in the operation code position of the No Operation instruction by the program. The next time this instruction is encountered in the program, program control will branch to the instruction stored starting at location XXX.)
BRU 1 XXX

Refer to BRU instruction.

Operation Name: Add

Mnemonic Code: ADD

Operation Code: 1

Instruction Format:

O M N AAA BBB CCC

O: Operation code 1

M: Length of AAA (1-12)¹

N: Length of BBB (1-12)¹

AAA: MSD (most significant digit) of augend

BBB: MSD of addend

CCC: MSD of sum

Add algebraically the numeric field at location AAA to the numeric field at location BBB, and store the results in location CCC.² The length of CCC will be the same as AAA or BBB, whichever is larger. Any carry greater than the length of CCC will be lost. The contents of locations AAA and BBB will be undisturbed unless the AAA or BBB address is the same as the CCC address. This instruction will set the comparison toggles to less than zero or minus, zero, or greater than zero or plus, depending on the result of the addition. Minus zero is nonexistent.

¹ A zero or blank in either the M or N position denotes a field length of 12. The # denotes a field length of 10; the @ denotes a field length of 11.

² If alphabetic or special characters are contained in the fields specified by AAA and BBB, only the numeric bits are considered. The zone bits of the least significant digit positions of fields AAA and BBB determine the sign of field CCC.

Note that in using the Add instruction the field length of CCC must be large enough to eliminate the possibility of the loss of the overflow unless the loss of the overflow is desirable.

Examples:

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>	<u>TOGGLE</u>
ADD	4	4	100	200	300	<u>SETTINGS</u>
Contents of				<u>Before</u>	<u>After</u>	
field starting at			100	0333	0333	
			200	0121	0121	
			300	0000	0454	Plus
			100	8324	8324	
			200	7457	7457	
			300	3333	5781*	Plus
			100	0028	0028	
			200	0343	0343	
			300	2222	0315	Plus

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>	
ADD	3	2	100	200	300	
Contents of				<u>Before</u>	<u>After</u>	
field starting at			100	028	028	
			200	62	62	
			300	888	034	Minus
			100	923	923	
			200	88	88	
			300	000	011*	Plus

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>	
ADD	3	3	100	200	200	
Contents of				<u>Before</u>	<u>After</u>	
field starting at			100	534	534	
			200	224	758	Plus
			100	735	735	
			200	324	059*	Plus
			100	028	028	
			200	320	292	Plus
			100	420	420	
			200	050	370	Minus

* Overflow lost due to insufficient field length.

Operation Name: Subtract

Mnemonic Code: SUB

Operation Code: 2

Instruction Format

O M N AAA BBB CCC

O: Operation code 2

M: Length of AAA (1-12)¹

N: Length of BBB (1-12)¹

AAA: MSD of minuend field

BBB: MSD of subtrahend field

CCC: MSD of difference field

Subtract algebraically the contents of the numeric field at location BBB from the contents of the numeric field at location AAA, and store the results in location CCC.² The length of CCC is the length of M or N, whichever is greater. Any carry greater than the length of CCC will be lost. The contents of locations AAA and BBB will be undisturbed unless the

AAA or BBB address is the same as the CCC address. This instruction will set the comparison toggles to minus, zero, or plus depending on the result of the subtraction. Minus zero is nonexistent.

Note that in using the Subtract instruction the field length of CCC must be large enough to eliminate the possibility of the loss of the overflow unless the loss is desirable.

Examples:

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>	<u>TOGGLE</u>
SUB	5	5	100	200	300	<u>SETTINGS</u>
Contents of field starting at						
			100	<u>Before</u>	<u>After</u>	
			200	02315	02315	
			300	01122	01122	
				99999	01193	Plus
			100	83724	83724	
			200	30324	30324	
			300	00000	14048*	Minus
			100	00303	00303	
			200	00248	00248	
			300	14048	00551	Plus

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>	<u>TOGGLE</u>
SUB	3	3	100	200	100	
Contents of field starting at						
			100	<u>Before</u>	<u>After</u>	
			200	200	150	
			200	050	050	Plus
			100	755	188*	
			200	433	433	Minus
			100	360	680	
			200	320	320	Plus
			100	234	355	
			200	121	121	Minus
			100	423	048	
			200	375	375	Minus

* Overflow lost due to insufficient field length.

Operation Name: Multiply

Mnemonic Code: MUL

Operation Code: 3

Instruction Format:

O M N AAA BBB CCC

O: Operation code 3

M: Length of AAA (1-12)¹

N: Length of BBB (1-12)¹

AAA: MSD of multiplicand field

BBB: MSD of multiplier field

CCC: MSD of product field

Multiply algebraically the contents of the numeric field at location AAA by the contents of the numeric field at location BBB, and store the result or product in the field at location CCC,² the length of which is M plus N. This instruction will set the comparison toggles to minus, zero, or plus as a result of the multiplication. Multiplication by a negative number which results in a minus zero can occur. A Branch instruction will cause program control to branch on zero, not minus.

Examples:

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>	<u>TOGGLE</u>
MUL	3	2	100	200	300	<u>SETTINGS</u>
Contents of field starting at				<i>Before</i>	<i>After</i>	
			100	536	536	
			200	18	18	
			300	82233	09648	Plus
			100	810	810	
			200	63	63	
			300	00000	51030	Minus
			100	222	222	
			200	44	44	
			300	33333	09768	Plus

Operation Name: Divide

Mnemonic Code: DIV

Operation Code: 4

Instruction Format:

O M N AAA BBB CCC

O: Operation code 4

M: Length of AAA (1-12)¹

N: Length of BBB (1-12)¹

AAA: MSD of dividend field

BBB: MSD of divisor field

CCC: MSD of quotient field

Divide algebraically the contents of the numeric field at location AAA by the contents of the numeric field at location BBB and store the result in location

CCC,² the length of which is M minus N. The remainder is stored in location AAA, right justified. Division by zero will result in a quotient of zero and the dividend will not be affected. Any time the absolute value of the divisor field is less than the absolute value of the corresponding high order digits of the dividend, the quotient will be zero and the dividend altered. If M minus N is equal to zero or negative, the machine will halt prior to executing the instruction, and will display the operation code.³

This instruction will set the comparison toggles to minus, zero, or plus as a result of the division. A minus zero quotient can occur on an illegal divide but the Compare-Branch instruction will cause program control to branch on zero, not minus.

³ See Section 5 for a description of the Data Processor control panel.

Examples:

<u>OP</u> DIV	<u>M</u> 4	<u>N</u> 2	<u>AAA</u> 100	<u>BBB</u> 200	<u>CCC</u> 300	<u>TOGGLE</u> <u>SETTINGS</u>
Contents of field starting at			100	<i>Before</i> 0326	<i>After</i> 0014	
			200	24	24	
			300	00	13	Plus
			100	2326	*	
			200	24	24	
			300	44	00	Zero
			100	1486	0034	
			200	44	44	
			300	85	33	Plus
			100	1422	1422	
			200	00	00	
			300	88	00	Zero

*This is an example of an illegal divide. The quotient is zero and the dividend is altered. To divide properly, the MSD of the dividend should be zero, thus making the division legal, as shown in the next example.

<u>OP</u> DIV	<u>M</u> 5	<u>N</u> 2	<u>AAA</u> 100	<u>BBB</u> 200	<u>CCC</u> 300	<u>TOGGLE</u> <u>SETTINGS</u>
Contents of field starting at			100	<i>Before</i> 02326	<i>After</i> 00022	
			200	24	24	
			300	444	096	Plus

Operation Name: Compare-Branch Equal

Mnemonic Code: CPA, CPZ, CPN

Operation Code: 5

Instruction Format:

O M N AAA BBB CCC

O: Operation code 5

M: 0 Alpha compare (CPA)

1 Zone compare (CPZ)

2 Numeric compare (CPN)

N: Lengths of AAA and BBB (1-12)

AAA: MSD of compared field

BBB: MSD of compared field

CCC: MSD of Branch address

Compare the contents of location AAA to the contents of location BBB and adjust the comparison toggles accordingly. If the contents of location AAA are less than the contents of location BBB the toggles are set to low. If the contents of location AAA are equal to the contents of location BBB the toggles are set to equal. If the contents of location AAA are greater than the contents of location BBB the toggles are set to high. If the comparison is

equal, program control will branch to the instruction starting at location CCC,⁴ otherwise, program control will continue in sequence.

M is the variant which determines the class or type of comparison:

If M equals 0, comparison is made on the alpha-numeric character.

If M equals 1, comparison is made on the zone bits only. In this case the collating sequence in ascending order is 12 zone (A bit), 11 zone (B bit), zero zone (A and B bits) and no zone (no A or B bits).

For example, on a zone compare, if the A and B bits were on in location AAA and just the A bit was on in location BBB, the comparison toggles would be set to high.

If M equals 2, comparison is made on numeric bits only.

The equal status of the comparison toggles will not be disturbed if the branch is taken. The numeric comparison is absolute and signs will be ignored. The results of the alpha comparison are determined by the collating sequence found in Appendix C.

⁴ Recall that the operation code of an instruction is always stored in the zero (most significant) digit position of a 12-character field. Therefore the third character position of a Branch instruction address will always be zero.

Examples:

Alphanumeric Comparison:

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>	<u>TOGGLE</u>
CPA	0	2	100	200	XXX	<u>SETTINGS</u>

Contents of field starting at	<u>100</u>	<u>200</u>	<u>Branch</u>	
	AB	AB	Yes	Equal
	12	AB	No	High
	AB	JK	No	Low

Zone Comparison Only:

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>
CPZ	1	1	100	200	XXX

Contents of field starting at	<u>100</u>	<u>200</u>	<u>Branch</u>	
	A	G	Yes	Equal
	A	1	No	Low
	9	8	Yes	Equal
	B	K	No	Low
	J	K	Yes	Equal

Numeric Comparison Only:

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>
CPN	2	3	100	200	XXX

Contents of field starting at	<u>100</u>	<u>200</u>	<u>Branch</u>	
	222	223	No	Low
	ABC	123	Yes	Equal
	ABC	ABB	No	High
	000	000	Yes	Equal
	JKL	JKM	No	Low
	JKL	ABC	Yes	Equal
	0JK	1S	Yes	Equal

(Refer to Appendix C for internal code structure.)

Operation Name: Branch

Mnemonic Code: BRC, BRU

Operation Code: 6

Instruction Format:

O M N AAA BBB CCC

O: Operation code 6

M: 0—Conditional Branch (BRC)

1—Unconditional Branch (BRU)

N: Not used

AAA: BRC—Branch address if low or minus

BRU—Branch address

BBB: BRC—Branch address if zero or equal

BRU—Not used

CCC: BRC—Branch address if high or plus

BRU—Not used

The Branch instruction with the operation code of 6 is used in two different ways.

When M is 0, the conditional branch calls for changing program control depending on the setting of the comparison toggle.

If the comparison toggles are set to low after a Compare instruction or minus after an arithmetic instruction, program control branches to the instruction starting at location AAA; if set to equal after a Compare instruction or zero after an arithmetic instruction, program control branches to the instruction starting at location BBB; if set to high after a Compare instruction or plus after an arithmetic instruction, program control branches to the instruction starting at location CCC. The Branch-Conditional

instruction should immediately follow the arithmetic or Compare instruction upon which the decision to branch is to be made. Other instructions affect the comparison toggles and may alter their setting.

When M equals 1, the instruction is an unconditional branch. Program control automatically branches to the instruction starting at location AAA. BBB and CCC are not used.

The comparison toggles are not affected by this instruction.

Operation Name: Transfer

Mnemonic Code: TFR

Operation Code: 7

Instruction Format:

O M N AAA BBB CCC

O: Operation code 7

M: Number of 12 character fields (0-9)

N: Number of remaining characters (0-11)

AAA: MSD of field(s) of information to be transferred

BBB: Not used

CCC: MSD of field(s) where information is to be transferred

Relocate alphanumeric data in core storage. Data starting with the contents of location AAA are transferred to another storage area starting with location CCC. The number of characters moved is designated by M and N, where M specifies the number of 12-character fields and N specifies the number of characters of a partial field. The maximum number of characters moved by a transfer instruction is 120. Zero or blank in the M or N positions denotes zero except when M and N are both zero, in which case 120 successive characters are transferred. This instruction does not affect the comparison toggles.

Example:

To relocate 55 characters, starting with the content of location 400, to another storage area starting with location 600, the transfer instruction would be written as:

7 4 7 400 XXX 600

Starting with the character in location 400 and moving from left to right, the instruction will move 4

fields of 12 characters each plus the next 7 characters for a total of 55 and will place these characters starting at location 600, again moving from left to right.

Operation Name: Mask

Mnemonic Code: MSK

Operation Code: 8

Instruction Format:

O M N AAA BBB CCC

O: Operation code 8

M: Length of AAA (1-12)¹

N: Not used

AAA: MSD of field to be masked

BBB: MSD of mask constant

CCC: MSD of masked field

Move the data starting with the contents of location AAA to a new storage area starting with location CCC, while inserting commas, decimals or credit symbol, and/or performing zero suppression controlled by the constant mask located in location BBB. The number of digits in the field to be masked is designated by M. Zero or blank in the M position denotes 12. The length of CCC and BBB is M plus the number of inserts; that is, the length of the original field at AAA as specified by M is increased by the number of inserts specified in the mask field at BBB as it is stored at CCC.

Zero suppression is always operative during this instruction. When high order zeros appear, a character in the Mask field will automatically be inserted until a significant digit in the AAA field is reached.

The character inserted can be a blank for zero suppression, an asterisk for check protection, or zero to override zero suppression. When high order zeros are encountered, a comma in the mask field will be replaced by the previously inserted character. A decimal point in the mask establishes significance and turns off zero suppression.

Other characters can be used if available on the printer installed with the system. After the last number in the field has been masked, automatic comparison of that number determines what is inserted in the character position immediately to the right of the masked field. If that number is negative, the low order position of the mask is inserted. Otherwise a blank is inserted. The comparison toggles are set to zero or equal at completion of this instruction.

Note: The contents of AAA and BBB are *not* affected by the mask command. Examples below illustrate the contents of AAA and BBB as they would appear before and after execution and the contents of CCC after execution.

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>
MSK	6		400	500	600
Contents of					
field starting at					
			400	294736	
			500	, .00-	
			600	2,947.36	
			400	003472̄	
			500	, .00-	
			600	34.72-	
			400	000785	
			500	0,000.00	Override zero suppression
			600	00007.85	
			400	006248	
			500	\$, .00	Insert dollar sign
			600	\$ 62.48	
			400	028735	
			500	*,***.00	Check Protection
			600	**287.35	

Operation Name: Halt

AAA: Not used

Mnemonic Code: HLT

BBB: Not used

Operation Code: 9

CCC: Not used

Instruction Format:

O M N AAA BBB CCC

O: Operation code 9

M: Halt identification number

N: HLT—Light 1 of 10 Indicators on Record Processor Console (1-10)

The system halts after all operations in progress have been completed. The Halt indicator on the Record Processor Console, corresponding to N, goes on. The operation code and the M and N variants are displayed on the Data Processor Console. The comparison toggles are not affected by this instruction.

Examples:

<u>OP</u>	<u>M</u>	<u>N</u>	<u>AAA</u>	<u>BBB</u>	<u>CCC</u>
HLT	0	4			

The system halts and the indicator on the console of the Record Processor corresponding to N goes on.

Operation Name: Card Read

Mnemonic Code: CRD

Operation Code: #

Instruction Format:

O M N AAA BBB CCC

O: Operation code #

M: Not used

N: Must be one (1)

AAA: Not used

BBB: Branch Address for End of File

CCC: MSD of input area

Read the 80 columns of information stored in the card buffer into the 80 positions of core storage starting at location CCC. The next card in the Program Card Reader hopper is automatically fed and the 80 columns of information punched in this card is read into the card buffer.

When a card has been incorrectly read, as determined by the read-in checking circuitry, the system halts on the next Card Read instruction and the buffer contents are not transferred to storage.

When a Card Read instruction is to be executed and the buffer has not been refilled due to an empty hopper, the system idles. Depression of the End of File key causes program control to branch to the instruction starting at location BBB. If it is not the end of the file, refill the hopper, then push the Start button on the Program Card Reader and the Continue button on the Data Processor. The comparison toggles are set to zero or equal on this instruction.

Operation Name: Sorter Read, Demand

Mnemonic Code: SRD

Operation Code: #

Instruction Format:

O M N AAA BBB CCC

O: Operation code #

M: Not used

N: Must be six (6)

AAA: Branch address for "can't read"

BBB: Branch address for End of File

CCC: MSD of Sorter input area

Transfer information stored in the sorter buffer to core storage. The first location of the input area is designated by CCC. Refilling the buffer is accomplished automatically by the Control Sorter instruction.

When a "can't read" condition occurs, a binary coded 15 is inserted in place of the invalid character(s) and program control branches to the instruction stored in location AAA.

End of file for the Sorter-Reader in the demand mode is determined by encountering a read instruction when the sorter buffer is empty due to an empty hopper and the system idles. Two courses of action can be taken depending on the conditions which caused the end of file.

1. Refill the sorter hopper and depress the Continue button to continue processing.
2. Depress the End of File button and program control will take the end of file branch to the instruction starting at location BBB for the appropriate routine.

The comparison toggles are set to zero or equal on this instruction. Word formatting of MICR information in storage is covered in Appendix A.

Operation Name: Sorter Read, Flow

Mnemonic Code: SRF

Operation Code: #

Instruction Format:

O M N AAA BBB CCC

O: Operation code #

M: Not used

N: 4—Read entire document

5—Stop reading at ending transit symbol

AAA: Branch address for "can't read"

BBB: Branch address for end of file

CCC: MSD of sorter input area

Process MICR documents at flow speed of the sorter, which is up to 1560 items per minute for minimum size documents of 5 $\frac{3}{4}$ inches in length. The characters read from these documents by the Sorter-Reader are transferred directly to core storage. The input area starts at location CCC.

When a "can't read" condition occurs a binary coded 15 is inserted in place of the character(s) that can't be read and program control branches to the instruction stored at location AAA.

When the Sorter Read, Flow instruction is not present as the leading edge of the item passes the read head, a stop flow is initiated and this item plus other items in the aligning station will be directed to the reject pocket. The system will halt on the next sorter instruction.

The comparison toggles are set to zero or equal on this instruction.

Word formatting of MICR information in storage is covered in Appendix A.

When a read instruction is encountered and no information has been transferred from a document to the Sorter-Reader read head, Sorter-Reader end of

file is determined. If a stop flow has been initiated either from execution of the Stop Flow instruction or from a signal sent from the batch ticket detecting device of the Sorter-Reader and the items in the transport have been processed, program control will take the end of file branch to the instruction starting at location BBB for the appropriate routine. If the system halts because no document has been fed, two courses of action can be taken depending on the conditions which caused the halt.

1. Refill the sorter hopper and depress the Continue button to continue processing.
2. Depress the End of File button and program control will take the end of file branch (BBB).

Operation Name: Control Sorter

Mnemonic Code: CTL

Operation Code: C

Instruction Format:

O M N AAA BBB CCC

O: Operation code C

M: Determines type of control

0—Pocket Select only

2—Demand feed one item and pocket select

4—Stop flow mode and pocket select

6—Start or continue flow and pocket select

N: Select pocket (0-11, 15)⁵

AAA: Not used

BBB: Not used

CCC: Not used

Control the sorter according to the variant M and select the pocket designated by N. This instruction only controls the feeding, pocket selecting and ejecting of items. When M is 2, one item will be fed, transferred to the buffer, and stopped at the stand-by station. A pocket will be selected and the item previously at the stand-by station sent to that pocket. When M is 6, the flow of items is either started or continued. Information from the items is transferred directly to storage by execution of a Read Sorter instruction.

In the demand mode a Control Sorter instruction cannot be executed when the buffer is still full. The system will halt. A read instruction must be given prior to each Control Sorter instruction.

In the flow mode, if a Control Sorter instruction is not executed within 29 ms. after the start of a Read Sorter instruction, the sorter will shut down and the system will halt.

For each item fed by the sorter, one Read Sorter and one Control Sorter instruction must be executed or the system will halt.

The comparison toggles are set to zero or equal on this instruction.

Operation Name: Record Processor Read

Mnemonic Code: RPR

Operation Code: G

Instruction Format:

O M N AAA BBB CCC

O: Operation code G

M: 1—Primary station

2—Manual station

3—Auxiliary station

N: Not used

AAA: Branch address on read error or feed of multiple documents

BBB: Not used and not available for storing extra data

CCC: MSD of input storage area

Pick a record from the station designated by M, feed it and read 81 characters of information from the magnetic stripe. Store the information in an input area starting with location CCC. The 81st character is a special character called a group mark (\equiv) which will signal the end of the read. The group mark is stored.

In the event of a read or double document error the records will be directed to the auxiliary stacker and the system will halt. Depression of the Continue button on the Console causes program control to branch to the instruction starting at AAA for the appropriate error routine.

When M is 4 or 5 this instruction can perform additional operations which utilize the N, AAA, BBB, and CCC character positions for different purposes than above.

M: 4—Auxiliary station and photo sensing

N: Not used

AAA: Not used

BBB: Not used

CCC: MSD of input storage area

Pick a record from the auxiliary station. Read the two digits of information (bar code) encoded on the face of the record into storage locations CCC and CCC + 1.

⁵ The 15 for-pocket select is designated by a special character (\equiv) for machine language coding.

M: 5—Keyboard

N: Light indicators on Keyboard (1-10)

AAA: Not used

BBB: Not used

CCC: MSD of input storage area

Read 13 digits indexed on the Keyboard (12 numeric and one special data) into a storage area starting with location CCC. The instruction will also light the Keyboard indicator corresponding to N. (See Entry Control Key, Section 5.)

The comparison toggles are set to zero or equal on this instruction.

Operation Name: Record Processor Write and Eject

Mnemonic Code: RPW

Operation Code: G

Instruction Format:

O M N AAA BBB CCC

O: Operation code G

M: 8—Eject and write

N: 0—Primary stacker
1—Auxiliary stacker

AAA: MSD of storage area

BBB: Not used

CCC: Branch address on write error

Eject the record in the print station to the stacker designated by N and write the information starting from location AAA in the record stripe. If a write error condition occurs, the record is directed to the auxiliary stacker and the system halts. Depression of the Continue button causes program control to branch to the instruction starting at location CCC for the proper error routine. Eighty-one characters are written from storage to the magnetic stripe. The 81st character is the group mark which signals the end of the information to be written.

M: 9—Eject only

N: 0—Primary stacker
1—Auxiliary stacker

AAA: Not used

BBB: Not used

CCC: Not used

Eject the record in the print station to the stacker designated by N without storing new information in the record stripe. The eject instruction must be preceded by an align or print instruction.

The comparison toggles are set to zero or equal on this instruction.

Operation Name: Record Processor Print

Mnemonic Code: PRV

Operation Code: H

Instruction Format:

O M N AAA BBB CCC

O: Operation code H

M: Not used

N: 0—Space suppress

1—Space form A

2—Space form B

3—Space forms A and B

4—Space record

5—Space record and form A

6—Space record and form B

7—Space record, forms A and B

AAA: MSD of storage area printed from

BBB: Branch address for form B last line

CCC: Branch address on print error

Print one line of 156 characters from 160 locations in storage starting with the character in location AAA. Four positions will be occupied by dummy print bars for the purpose of supporting the ribbon. Blanks must be inserted in storage corresponding to these positions. Single space the record and forms, after printing, according to variant N.

If during execution of this instruction the last line is printed on form B, program control branches to the instruction starting at location BBB.

When a print instruction is to be executed and the line find mechanism has failed to operate, the system will halt and the Align Check indicator on the console goes on. The record proceeds to the primary stacker. Depression of the Continue button will cause the next instruction in sequence to be executed.

If a print error occurs, the system halts and spacing is suppressed. When the Continue button is depressed, program control transfers to the instruction starting at location CCC. A branch on print error takes precedence over a form B last line branch.

At the completion of the print instruction the comparison toggles are in the zero or equal state.

Operation Name: Skip Record Processor

Mnemonic Code: SKR

Operation Code: I

Instruction Format:

O M N AAA BBB CCC

O: Operation code I

M: 1—Restore forms only

2—Single space forms and ledger according to variant N

4—Align record

N: 1—Space form A

2—Space form B

3—Space forms A and B

4—Space ledger

5—Space ledger and form A

6—Space ledger and form B

7—Space ledger and forms A and B

AAA: Address of MSD of line find digits when M is 4. Otherwise not used.

BBB: Branch address of Form B last line.

CCC: Not used

Controls form spacing or skipping on the Record Processor according to variants M and N.

When M is four, the last record fed will be stopped in the printer at the print line (0-47) specified by the line find information stored starting at location AAA. This information is either previously read from the magnetic strip for storage or is prestored.

If an align instruction is given too late to stop the record in the print station, the align instruction will not be executed. The system will halt and the record will go directly to the primary stacker. The Align Check indicator will light.

At completion of this instruction the comparison toggles are set to zero or equal.

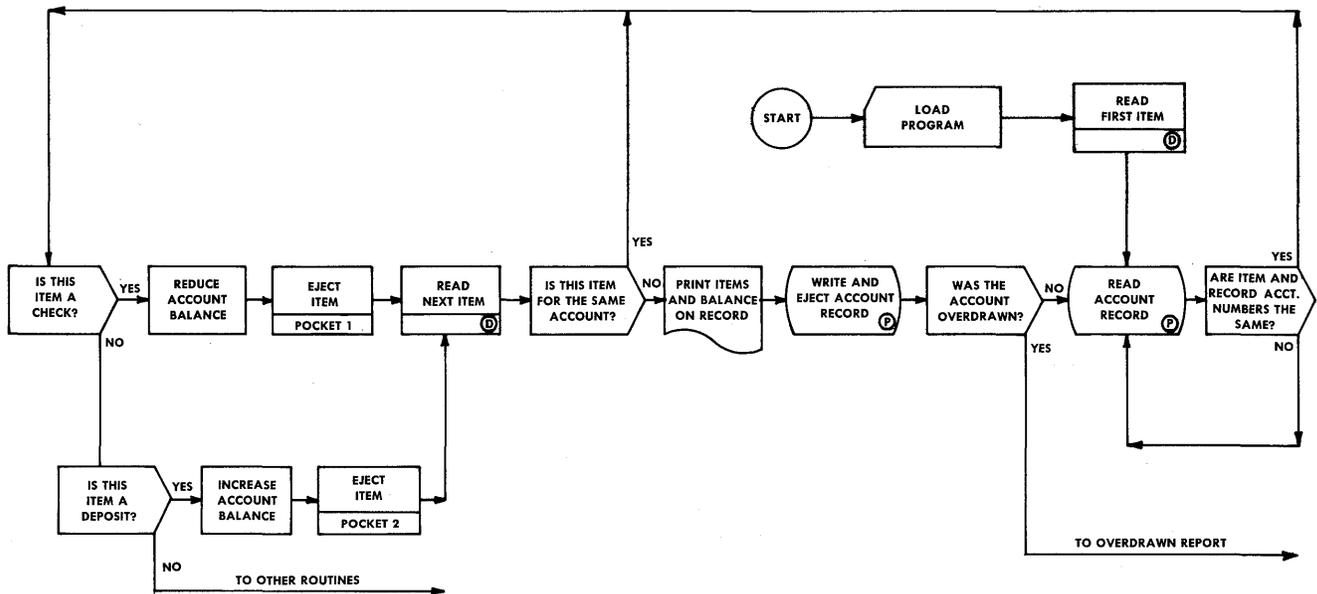
SAMPLE PROBLEM

For a sample program using most of the instructions, a simple bank account updating application was selected. With the program is the flow chart from which the program was developed, a list of the storage location assignments, a memory map showing the individual assignment in storage, and the input and output layout forms.

A simple addressing system was employed in the sample for clarity and better understanding. The three characters of the address, separated by deci-

mal points, represent the section, field, and character, respectively, of the location. For example, 7.1.5 means the 5th character location in the first field of the seventh section.

It is assumed in the sample that the transaction code on a check is blank and on a deposit is one. It was arbitrarily decided to use XXX for the branch addresses for can't read, end of file, read error, invalid transaction code, other item routine, form B last line, print error, write error, and overdrawn account routine.



Flow Chart for Sample Banking Application

LOCATION ASSIGNMENT

Instructions		0 0.0
Item Input Area		3 0.0
Account number, M.S.D. — Length - 7	3 4.5	
Transaction Code, M.S.D. — Length - 1	3 5.@	
Amount, M.S.D. — Length - 10	3 6.1	
Account Record Input Area		4 0.0
Line Final Data, M.S.D. — Length - 2	4 0.0	
Account number, M.S.D. — Length - 7	4 0.2	
Balance, M.S.D. — Length - 12	4 1.0	
Printer Output Area		5 0.0
Checks, M.S.D.	5 2.5	
Deposits, M.S.D.	5 3.#	
Date, M.S.D.	5 5.3	
Balance, M.S.D.	5 6.5	
Blank Area for Clearing		6 0.0
Constant Area		7 0.0
Mask, M.S.D.	7 0.0	
Date, M.S.D. — Length - 7	7 1.5	

SAMPLE PROGRAM

LOC	MNE	OP	M	N	AAA	BBB	CCC	REMARKS
0.0.0	SRD	#		6	XXX	XXX	3.0.0	Read first item.
0.1.0	RPR	G	1		XXX		4.0.0	Read account record.
0.2.0	CPN	5	2	7	4.0.2	3.4.5	0.4.0	Are item and record account number equal?
0.3.0	BRU	6	1		0.1.0			To read next record.
0.4.0	SKR	I	4		4.0.0			Align record specified by line find.
0.5.0	CPA	5	0	1	3.5.@	0.1.2	1.2.0	Is this item a check? T.C. = blank.
0.6.0	CPA	5	0	1	3.5.@	0.6.2	0.8.0	Is this item a deposit? T.C. = 1.
0.7.0	BRC	6	0		XXX		XXX	To error and other item routine.
0.8.0	MSK	8	#		3.6.1	7.0.3	5.3.#	Mask deposit amount to print area.
0.9.0	ADD	1	#	0	3.6.1	4.1.0	4.1.0	Increase balance.
1.0.0	CTL	C	2	2				Eject deposit to pocket 2.
1.1.0	BRU	6	1		1.5.0			To read next item.
1.2.0	MSK	8	#		3.6.1	7.0.3	5.2.5	Mask check amount to print area.
1.3.0	SUB	2	0	#	4.1.0	3.6.1	4.1.0	Reduce balance.
1.4.0	CTL	C	2	1				Eject check to pocket 1.
1.5.0	SRD	#		6	XXX	XXX	3.0.0	Read next item.
1.6.0	CPN	5	2	7	4.0.2	3.4.5	0.5.0	Is this item for same account?
1.7.0	MSK	8	0		4.1.0	7.0.0	5.6.5	Mask balance to print area.
1.8.0	TFR	7	0	9	7.1.5		5.5.3	Transfer date to print area.
1.9.0	PRV	H		5	5.0.0	XXX	XXX	Print items, date, and balance on record.
2.0.0	ADD	1	1	2	2.0.0	4.0.0	4.0.0	Increase line find.
2.1.0	RPW	G	8	0	4.0.0		XXX	Write and eject account record.
2.2.0	TFR	7	8	0	6.0.0		5.0.0	Blank out print area.
2.3.0	CPZ	5	1	1	4.1.@	0.1.0	XXX	Was the account overdrawn?
2.4.0	BRU	6	1		0.1.0			To read next record.

INPUT-OUTPUT LAYOUT

TITLE: _____ PROGRAM NO. _____ PROGRAMMER _____ DATE: _____ PAGE: _____ OF: _____

	0	1	2	3	4	5	6	7	8	9	
0											0
1											1
2											2
3											3
4		ITEM	INPUT	AREA	LAYOUT						4
5											5
6											6
7											7
8						ACCOUNT NUMBER	TRANSACTION CODE	AMOUNT			8
9						##XXXXXX	##X#XXXXXX	XXXXXX			9
A											A
B											B
C		ACCOUNT	RECORD	INPUT	OUTPUT	AREA	LAYOUT				C
D											D
E											E
F											F
G		LINE ACCOUNT									G
H		FIND NUMBER	BALANCE								H
I		LFXXXXXXXX	XXXXXXXXXXXX								I
J											J
K											K
L			PRINTER	OUTPUT	AREA						L
M											M
N											N
O											O
P			CHECK COLUMN	DEPOSIT COLUMN	DATE	BALANCE					P
Q											Q
R			XX,XXX,	XX,XXX,XXX.XX	MON DA YR	X,XXX,XXX,XXX.XX-					R
BK											BK
/											/
S											S
T											T
U											U
V											V
W											W
X											X
Y											Y
Z											Z

4-15

3

5

SECTION 5

SYSTEM COMPONENTS

The Burroughs B 251 Visible Record Computer System is made up of five component parts; the Data Processor, the Card Reader, the high-speed Sorter-Reader, the Record Processor, and the Console and Operator's Desk. The complete system is shown in the frontispiece.

B 101 SORTER-READER

The Burroughs B 101 Sorter-Reader is a precision electronic device that reads and sorts magnetically encoded documents at speeds as high as 1560 documents per minute. The Sorter-Reader is compact in design and easily operated.

When operating as a digital sorter for magnetically encoded documents, the Sorter-Reader is set to the sort mode of operation and functions under the control of its own logic and internal programming. (See the B 101 Manual for a description of the Sorter-Reader operation.) In this mode, items are passed through and read at the free flow rate of the Sorter-Reader. The Sorter-Reader sorts the items into pockets corresponding to the value of the digit read as selected by the field and digit position selection buttons. For this purpose, ten pockets 0 through 9, two special pockets X and Y, and a reject pocket are used.

When utilized as an input device for the B 251 System, the Sorter-Reader is set to the external mode of operation. In this mode, the Sorter-Reader is under supervision of the Data Processor. Items pass through and are read at a demand or flow rate. As each item passes the read station, the character information is translated into binary coded decimal form and transmitted to the Data Processor.

There is a device on the Sorter-Reader called a batch ticket detector. When a solid black area on the front of a batch ticket is detected by this device while the Sorter-Reader is in the flow mode, the flow of items will be stopped.

The pocket destination for each item is determined by the Data Processor, which has two-way com-

munication with the Sorter-Reader. The pocket selection is based on program decisions and the pocket designation is transferred to the Sorter-Reader for disposition of the item. Figure 5-1 depicts the item processing route of the B 101.

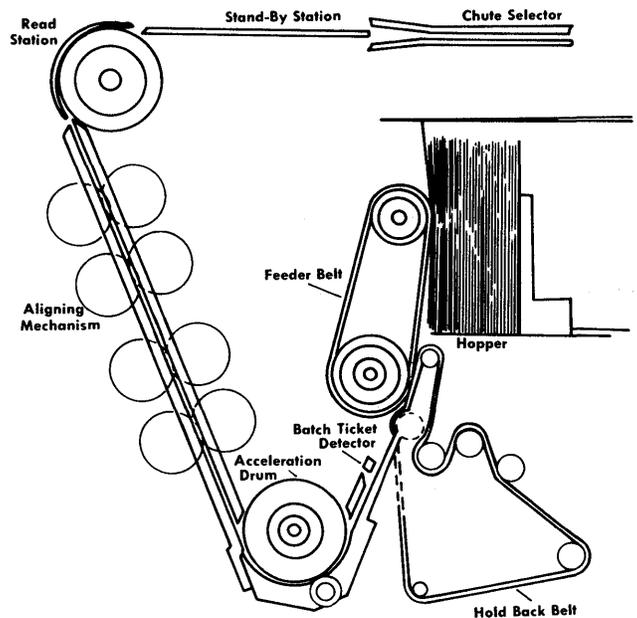


Figure 5-1. Sorter-Reader Item Processing Route

At this point, a few pertinent remarks concerning the B 101 Sorter-Reader in the external mode of operation are in order to familiarize the user with its function in relation to the B 251 Visible Record Computer System.

SORTER-READER CONTROL PANEL

On the control panel (Figure 5-2) of the B 101 Sorter-Reader there are four major groupings of operation buttons and indicators: communication indicators, operating buttons, field selection buttons, and digit position selection buttons. Only those

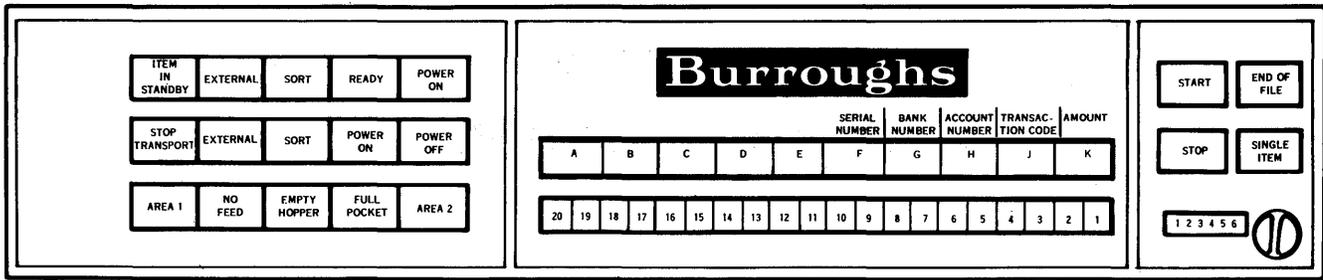


Figure 5-2. Sorter-Reader Control Panel

indicators and buttons that are not described fully in the B 101 Sorter-Reader Operator's Manual are included here.

The indicators on the control panel are lights which go on to signal the operator of conditions related to various control settings, document location, potential and actual stoppage conditions, and methods of control.

EXTERNAL. This green indicator is on whenever the Sorter-Reader is under control of the B 251 System. It indicates that the Sorter-Reader is being used for input and all supervisory control and pocket selection are being determined by the B 251 System, rather than the Sorter-Reader patchboard.

To engage the Sorter-Reader in the external mode, the operator must depress the button labeled External. This automatically transfers decision-making ability to the Data Processor.

ITEM IN STAND-BY. This red indicator is operational when the Sorter-Reader is in the external mode. It signals that an item has not been completely processed. It will flash as items pass through the stand-by station.

EMPTY HOPPER. This white indicator goes on when the hopper of the document feeder is empty; that is, the last document has been picked from the hopper. The system shuts down at this point and remains inoperative until the document feeder hopper is refilled and the Start button is depressed.

END OF FILE. Depressing this button causes program control to branch to an end of file subroutine. (See Read Sorter Demand and Read Sorter Flow instructions.)

DATA PROCESSOR

The Data Processor is a completely solid-state unit employing transistorized circuitry and housing the magnetic core storage or memory of 4800 alphanumeric character locations.

Basically, the Data Processor is the center of operations of the B 251 System. The Data Processor performs the logical operations, arithmetic, masking, checking, and editing. It also controls the operation of the input-output components of the B 251 System.

DATA PROCESSOR CONSOLE

The B 250 Data Processor provides a Console display panel (Figure 5-3) containing switches and indicators for operation of the system.

POWER ON-POWER OFF. In the upper left corner is a combination switch and indicator to turn power on. When the switch is depressed, the indicator goes on to show that power is on. Below it is a switch to turn off the power.

LOAD. Depressing this switch initiates a Card Read instruction which reads a series of 80 punched cards and stores the data from the cards sequentially in memory starting in location 000 and continuing until all 4800 positions of storage are filled and the Instruction Address register is again at 000. The instruction stored at 000 will be picked up and executed. Only the first 60 columns of each of the 80 cards are read and stored. The last 20 columns are available for identification and sequencing.

EMERGENCY POWER OFF. This is a pull type switch to shut down all power in case of an emergency. When this switch is pulled only a field engineer can restore the power. It is located in the lower left corner.

To the right of the power switches is a column of indicators, labeled Record Processor, Sorter, Card, and Data Processor, any or all of which go on whenever the corresponding unit of the system is in a not ready condition, for example, Record Processor power off, Sorter-Reader hopper empty, etc. These indicators will also go on whenever an error is detected by the corresponding unit of the system while reading, writing, printing, or feeding. A parity

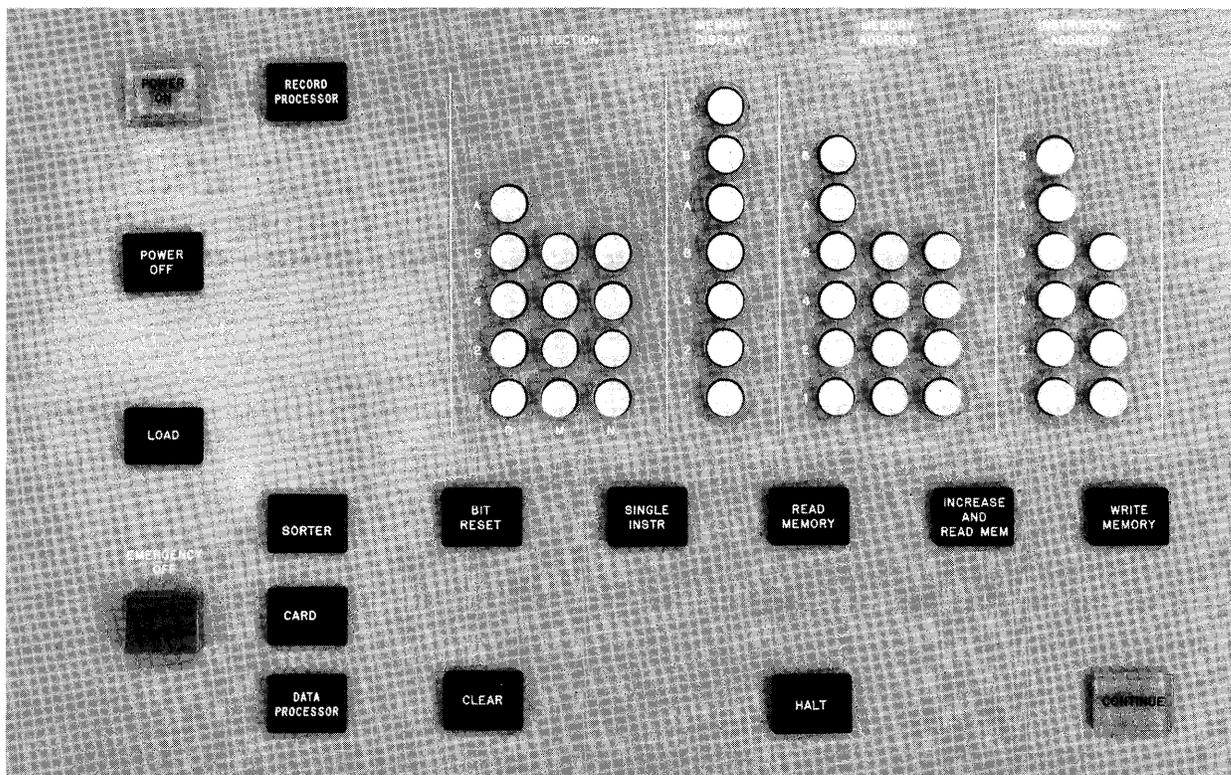


Figure 5-3. Data Processor Console

error will cause the Data Processor indicator to go on.

A bank of lights located in the upper right corner of the Console displays the contents of four registers. (A register is a temporary storage area or location for parts of instructions or pieces of data. These registers are primarily used for manual operations on the system.)

INSTRUCTION REGISTER. Three columns of lights which display the operation code and M and N variants of the instruction being executed.

MEMORY DISPLAY REGISTER. A column of seven lights which displays one character at a time, either one called from storage for display or change or a character manually inserted in the register for transfer to storage. The fact that the lights can be set or cleared manually and individually allows changing and inserting characters in the register.

MEMORY ADDRESS REGISTER. Three columns of lights which display a storage address. The address displayed is either that of the location of a character from storage to be transferred to the Memory Display register or that of a location in which a character from the Memory Display register is to be stored. These lights also can be set and cleared manually and individually.

INSTRUCTION ADDRESS REGISTER. Two columns of lights which display the address of the next instruction to be executed. Since the third character of a machine language instruction address is always zero, only the first two digits need be displayed. These lights also can be set and cleared manually and individually.

Directly below the registers is a horizontal row of switches which are used to manipulate the contents of the registers and to control the mode of operation.

BIT RESET. This switch allows clearing of individual bits in the Memory Display, Memory Address, and Instruction Address registers. By depressing a switch corresponding to a light in a register and the Bit Reset switch simultaneously, that light goes off.

SINGLE INSTRUCTION. Each time this switch is depressed, the instruction stored starting at the address displayed in the Instruction Address Register is executed.

READ MEMORY. When this switch is depressed, the character in the storage location indicated by the Memory Address register is displayed in the Memory Display register.

INCREASE AND READ MEMORY. Each time this switch is depressed, the contents of the Memory

Address register are increased by one and the character stored in the new location whose address is displayed in the Memory Address register is displayed in the Memory Display register.

WRITE MEMORY. When this switch is depressed, the character displayed in the Memory Display register is stored in the location indicated by the Memory Address register.

Below this row is another horizontal row of switches and lights.

CLEAR. Depressing the Clear switch clears (turns off) all indicators and sets all registers to zero.

HALT. This is a combination switch and indicator. When this switch is depressed the system will halt after all operations in progress have been completed. The indicator goes on when the system halts for any reason.

CONTINUE. This is a combination switch and indicator. Depressing this switch allows automatic operation to continue after a halt. If the halt is due to an error condition, depression of the switch will allow operation to continue after the error condition is removed.

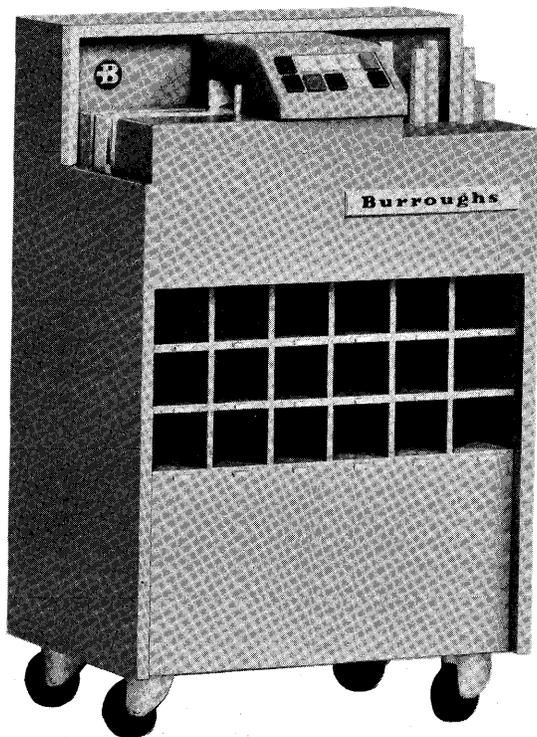


Figure 5-4. Program Card Reader

PROGRAM CARD READER

The Burroughs B 251 Visible Record Computer System utilizes a low-speed, on-line, tabulating card input device for loading programs into Data Processor storage. The Card Reader transport contains all the necessary components to pick a standard 80 column tabulating card from the hopper, feed it through the reading stations and finally direct it to the card stacker. The Card Reader is designed to handle cards cut on any of the four corners and cards notched on the right edge to indicate verification. A display panel is furnished with switches and indicators for operation of the Card Reader and to indicate error conditions.

PROGRAM CARD READER FUNCTIONAL OPERATION

HOPPER. The Card Reader provides a single hopper with a maximum capacity of 500 standard tabulating cards. Cards are placed in the hopper face down, column one first.

FEED. The feeding mechanism is a single feed and is used to transport cards from the picking mechanism to the data read station. Failure to feed causes a missing card signal to the computer. The Card Reader will halt with no more than two cards in a jammed condition.

READ. The data read station is a single, one-column data reader which reads cards serially column by column in tabulating card code. This code is translated into a six-bit binary coded decimal form (Appendix C). The data is transmitted to the computer in a parallel-by-bit, serial-by-character manner. Read checking is provided.

STACKER. The stacker stacks the cards in the sequence they were fed. The stacker has a maximum capacity of 500 cards. A not ready signal is emitted when the stacker reaches its capacity.

PROGRAM CARD READER DISPLAY PANEL

Switches are provided on the Program Card Reader for controlling its operation.

START. The Start switch is used to condition the card reader for feeding cards under control of the Data Processor at the start of a run and to restart after the hopper runs out of cards.

END OF FILE. The End of File switch is used when no more cards are to be processed after the hopper empties. When the Card Reader sends a signal to the Data Processor that the hopper is empty,

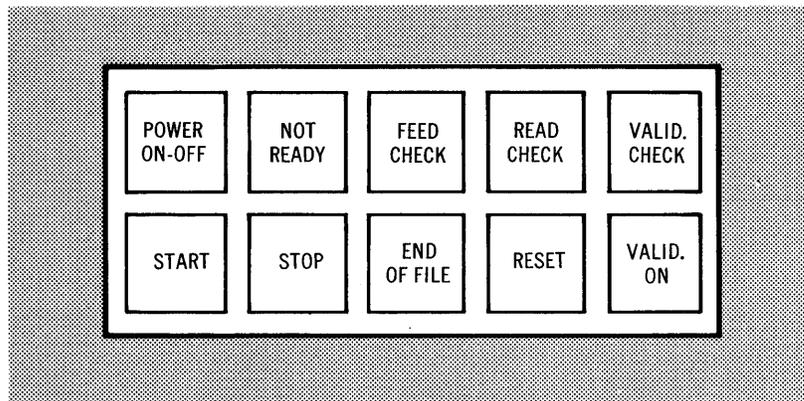


Figure 5-5. Program Card Reader Display Panel

the system will halt on the next Card Read command. By depressing the End of File switch, program control will automatically branch to the end of file routine specified by the Card Read command.

STOP. The Stop switch is used to stop the system at the completion of the command in operation.

RESET. The Reset switch is used to reset all error and not ready indicators.

POWER ON-OFF. The Power On-Off switch controls power for the Card Reader only.

VALIDITY ON. The Validity On switch provides checking for invalid characters read by the Program Card Reader. When the switch is off, validity checking is not performed.

Indicators are provided to show operation and error conditions.

NOT READY. The Not Ready indicator goes on when any of the not ready conditions exist. These conditions are: empty hopper, filled stacker, card jam, stop, failure to feed, and covers off.

FEED CHECK. The Feed Check indicator goes on as a result of a jam or a failure to feed.

READ CHECK. The Read Check indicator goes on when the read checking circuitry detects an operational failure.

VALIDITY CHECK. The Validity Check indicator goes on when an invalid character has been read by the Program Card Reader. The system halts. The Validity Check indicator and validity checking are only operative when the Validity switch is on.

THE RECORD PROCESSOR

The B 401 Record Processor is the automatic visible record processing unit of the Burroughs B 251 System. The Record Processor will automatically feed, read (from the record's magnetic stripe), print on, and eject the record document (Figure 5-6). The Record Processor is also capable of printing two, separately controlled, continuous forms. All three documents may be printed concurrently.

The Record Processor will accept visible record documents within a range (width to height) of 6½" x 11" to 14" x 11". A magnetic stripe on the back of each record can store 81 characters of digital information to identify the next available printing line, special types of visible records, or other data pertinent to the individual account. Three record feeders are provided. The feeding may be performed from a magazine in the primary station, which will hold up to a maximum of 1000 records, from the auxiliary station with a capacity of 500 records, or from an insertion slot for manual entry of one record. Two stations for reading the magnetic information are provided: one following the primary feed station and one following the auxiliary and manual feed stations.

An aligning station positions the record at the print line designated by a Data Processor instruction. Single spacing of the record is provided. When the record is ejected, new information can be recorded on the magnetic stripe in the recording station or the old information can be left undisturbed. A post read head is included at the recording station to check the recording operation. Two stackers are provided for the ejected records, the magazine in the Primary

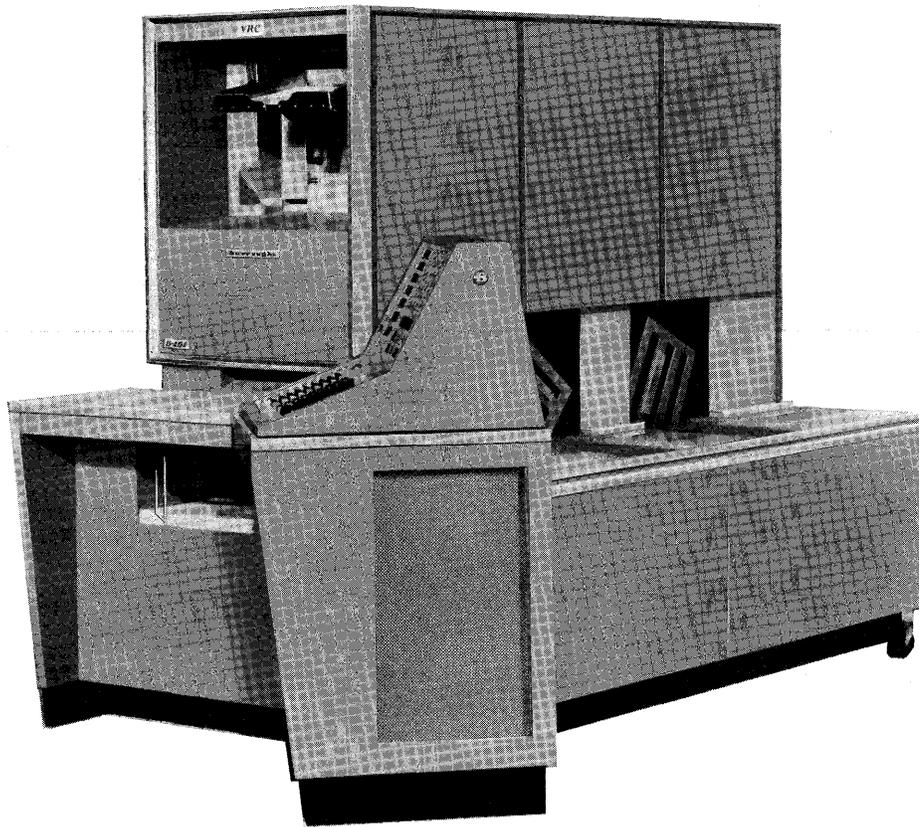


Figure 5-6. The Record Processor

Stacker with a capacity of 1000 records or an auxiliary stacker with a capacity of 150 records.

Continuous standard fan-fold forms up to a maximum width of 18 inches, which are margin-punched on both sides, may be printed in the Record Processor. Single spacing is provided and forms are automatically skipped across folds. A maximum of four sheets interleaved with three carbons can be handled simultaneously when no record is present. However, if records are being passed and printed simultaneously, only three sheets interleaved with two carbons can be handled.

The Record Processor contains 160 print positions spaced 10 to the inch. Standard type bars contain digits 0 to 9, plus a decimal and comma or credit symbol. Special type bars are available at the user's option.

RECORD PROCESSOR FUNCTIONAL OPERATION

RECORD TRANSPORT. The explanation of the record transport mechanism (Figure 5-7) is best defined by each of the following paragraphs describing the various record input and placement stations. The record transport moves records from the three input locations (primary feeder, auxiliary feeder, and man-

ual feeder insertion slot) to the printing station and, after printing, to either of two output stackers (primary stacker or auxiliary stacker).

PRIMARY FEEDER. The primary feeder is used to transport the visible records from the primary magazine to the primary ready station. The ready station is automatically reloaded from the primary feeder following a read command. The feeder contains an elevator to raise the magazine to the feeding position and automatically lower the magazine when empty. The feeder also contains a record width adjustment to determine picker location. All records for one run must be the same width; they are picked face up, top edge first. Capacity of one magazine is 1000 records.

AUXILIARY FEEDER. The auxiliary feeder is used to transport records to the auxiliary ready station from the elevator on which they are placed, face up, top edge first. No magazine is used. The elevator raises the records to the feeding position and automatically lowers when the last record is fed. The feeder contains a record width adjustment to determine picker location and guide the right edge of the record. Records must all be the same width during any one run. Capacity of the auxiliary feeder is 500 records.

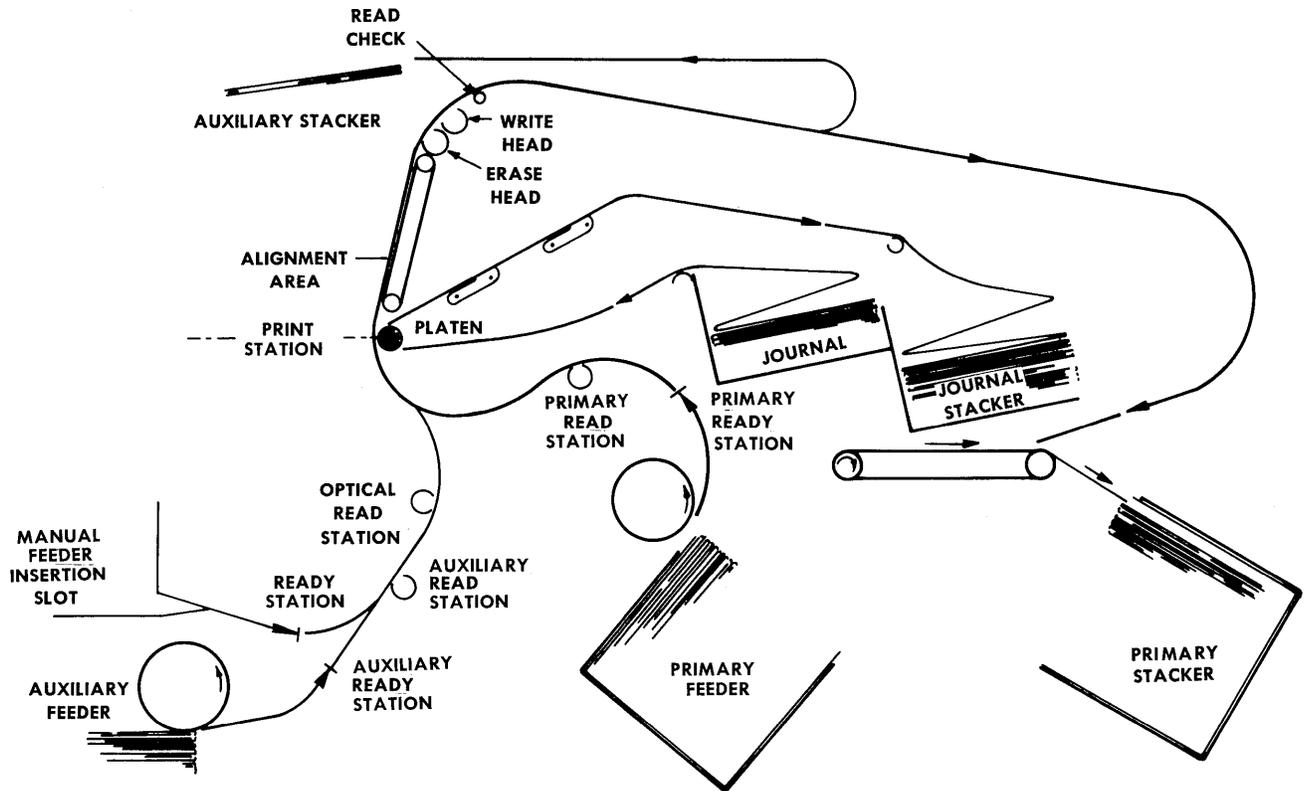


Figure 5-7. Record Processor Schematic

MANUAL FEEDER. The manual feeder has a capacity of one record and is used to transport the record to the ready station. A mechanical gate, electronically controlled by the Data Processor, prevents entry of subsequent records until information from the record in the manual ready station has been read by the Record Processor correctly. An indicator above the manual insertion slot is on when the ready station is not occupied.

READY STATIONS. There is a ready station for each of the three feeders. A record is held at the ready station until called for by a Record Processor read instruction. At this time the next record is automatically fed to the ready station.

PRIMARY READ STATION. The primary read station immediately follows the primary ready station in the record transport mechanism. Information recorded on the record magnetic stripe is read at the primary read station. Information is read serially and then stored in memory. Only records from the primary feeder are read at the primary read station.

AUXILIARY READ STATION. The auxiliary read station immediately follows the auxiliary and manual ready stations in the record transport mechanism. The function of the auxiliary read station is identical with that of the primary read station, except that

the auxiliary read station is used only for records fed from the auxiliary and manual feeders.

OPTICAL READER (OPTIONAL). An optical reader is available with the B 251 System. The optical reader reads the two binary coded decimal digits (bar code) printed on the face of a record. Detailed information concerning the optical reader will be furnished users upon request.

PRINTING UNIT. The Record Processor utilizes a printer containing 160 print positions spaced horizontally 10 to the inch. Each print position contains 12 characters. All type bars are interchangeable, in all positions, by the operator. Four type bars are blank and are used for the purpose of ribbon support. The maximum spacing between ribbon supports is 40 print positions.

The form movement mechanisms deliver forms A and B to a stacker where they are stacked side by side. Both input and output stackers will accommodate a stack of forms not exceeding 6 inches in height. The maximum length of a form will not exceed 11 inches fold to fold.

LINE FIND MECHANISM. The line find mechanism positions the correct line of the record in the printing station. The actual alignment signal originates in the Data Processor as a result of the SKR

instruction. Whenever an Eject instruction is delivered from the Record Processor, the line find mechanism releases the record from the print station.

FORM MOVEMENT MECHANISM. The Record Processor utilizes two continuous form feeding mechanisms, in addition to the visible record feed mechanisms described earlier. For practical purposes the two form feeders are called form A and form B mechanisms. Form tractors are provided to space continuous forms and automatically skip over folds. The width of forms A and B combined may not exceed 18 inches including sprocket holes. The minimum distance between the two forms is $\frac{3}{8}$ inch. The minimum width of a form is $3\frac{3}{4}$ inches including sprocket holes. A user will pattern form sizes to his individual needs. An example of different form sizes is illustrated in Figure 5-8.

When necessary, form size can be cut down to $3\frac{1}{4}$ inches and driven by only one tractor. The form cannot be overlapped by the record.

ERASE STATION. When a record is released from the printer it passes through the erase station to demagnetize the stripe. The erase head is operational only when new stripe information is to be written. It is located just ahead of the recording station.

RECORDING STATION. The recording station is the area where the actual writing of new information on the magnetic stripe of the record is done. The recording station is located following the erase station in the record transport mechanism.

CHECKING STATION. The checking station, immediately following the recording station, is used to determine reliability; that is, information from storage written upon the stripe is checked to insure that there has been a successful transfer and recording of information.

PRIMARY STACKER. The primary stacker receives records from the Record Processor transport mechanism and places them in the output magazine, face down, bottom edge first. The capacity of the

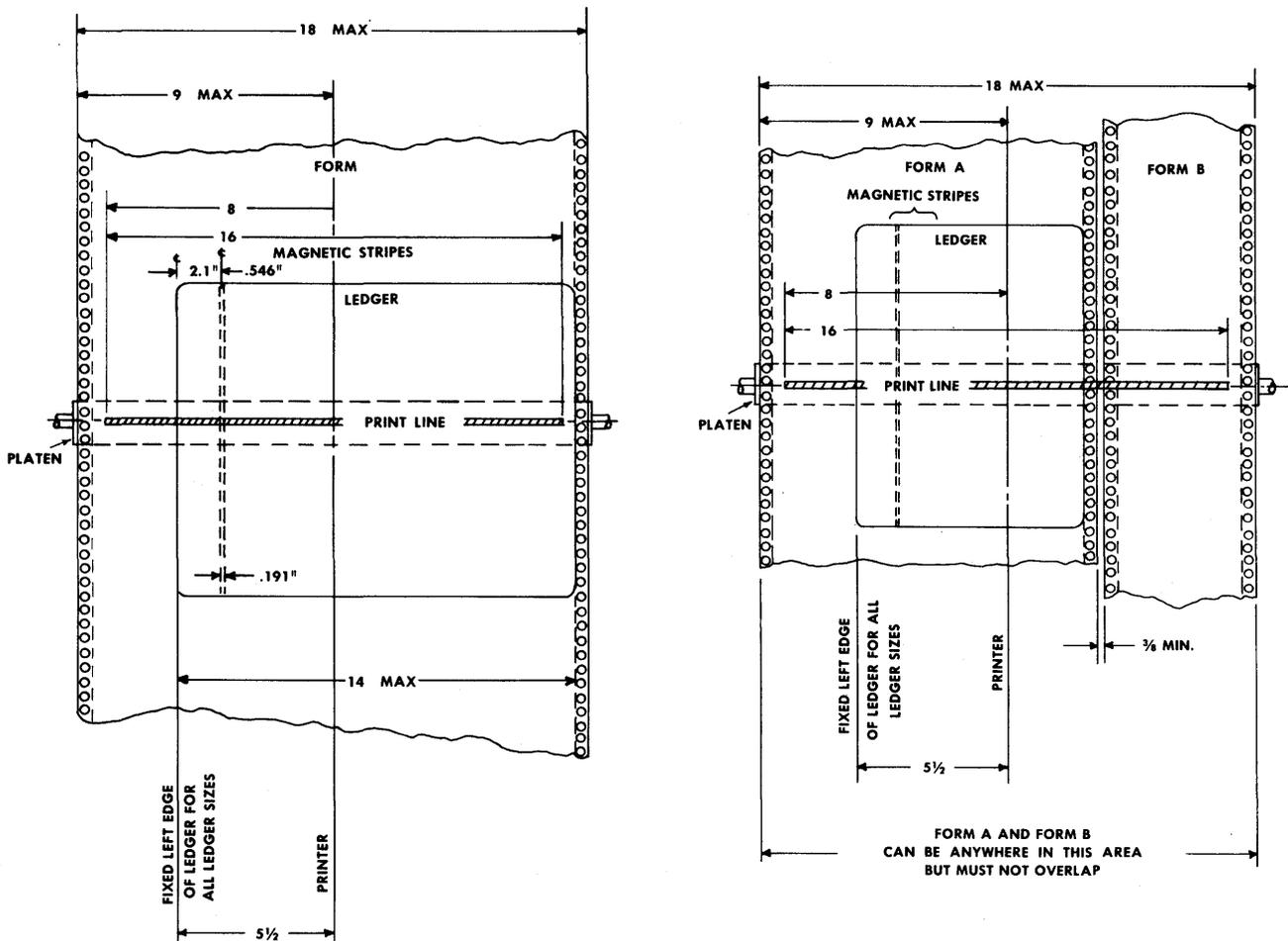


Figure 5-8. Examples of Different Form Sizes

primary stacker is 1000 records. When the primary stacker is filled to capacity, a signal is available which causes the entire system to shut down and remain inoperative until an empty magazine is inserted into position. For operator efficiency and convenience a magazine cart will be available with the B 251 System.

AUXILIARY STACKER. The auxiliary stacker accepts records from the record transport mechanism and places them in the stacker, face down, top edge first. The capacity of the auxiliary stacker is 150 records. When the stacker is filled, a signal is available which causes the entire system to shut down and remain inoperative until the stacker is emptied.

CONSOLE AND DESK

Attached to and part of the Record Processor is the Console and Operator's Desk (Figure 5-9). The Console provides the operator with the facilities necessary to initiate and control program operations.

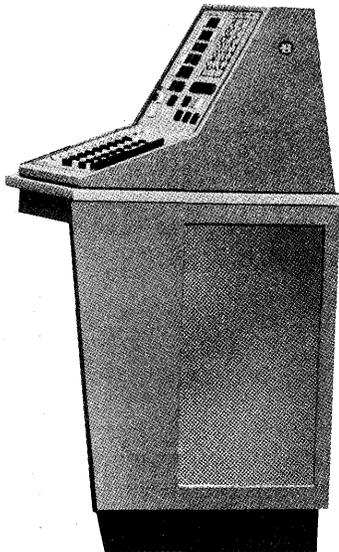


Figure 5-9. Console and Desk

CONSOLE CONTROL PANEL

The Console control panel (Figure 5-10) contains switches and indicators for system operation: power supply switches and indicators, indicators for the mode in which the system is operating, error condition indicators, and lights for displaying register contents.

Located in the left section of the Console control panel are four columns of indicators and switches.

The first (labeled Stops) is a column of indicators for input-output conditions. From top to bottom are the Primary Feeder, Auxiliary Feeder, and Manual Feeder indicators which are on during the execution of a read instruction from any of these stations. Next are the Form A and Form B indicators. These are on whenever their respective form supplies are almost exhausted. Below these are the Primary Stacker, Auxiliary Stacker, and In/Out indicators. The Primary Stacker and Auxiliary Stacker indicators go on when the stackers become full and the In/Out indicator is on during the execution of an input or output instruction.

In the second column (labeled Checks) is a series of indicators for error checks. Whenever the system halts due to a write error, print error, read error, feed error, parity error, or align error the indicator corresponding to the condition is on.

The third column is a series of switches for control of form and record movement. The top two switches are for spacing form A and form B. The next two are for restoring form A and form B to the home position. The fifth switch is for transporting any records that may be in the manual or auxiliary ready stations or the print station to the auxiliary stacker. The sixth switch in the column initiates raising of the magazine in the primary station to the feed position. The last switch in this column is for ejecting the record in the print station to the auxiliary stacker without writing new information on the magnetic stripe.

The last column contains only two indicators, System Not Ready and R. P. Not Ready.

SYSTEM NOT READY. This indicator is on when the B 251 System is unable to operate automatically as a system. The units capable of operation can be operated as a partial system.

R. P. NOT READY. This indicator is on whenever the Record Processor is not ready for operation.

In the lower left section of the Console control panel is a pull-type switch, called Emergency, that shuts down all system power. A field engineer is required to restore the power and to bring the system back to operational status.

In the upper right section of the control panel is a bank of lights for viewing the contents of the following display registers. Using the first two of these registers and the Write Memory switch (see below) information can be manually entered into core storage, a character at a time.

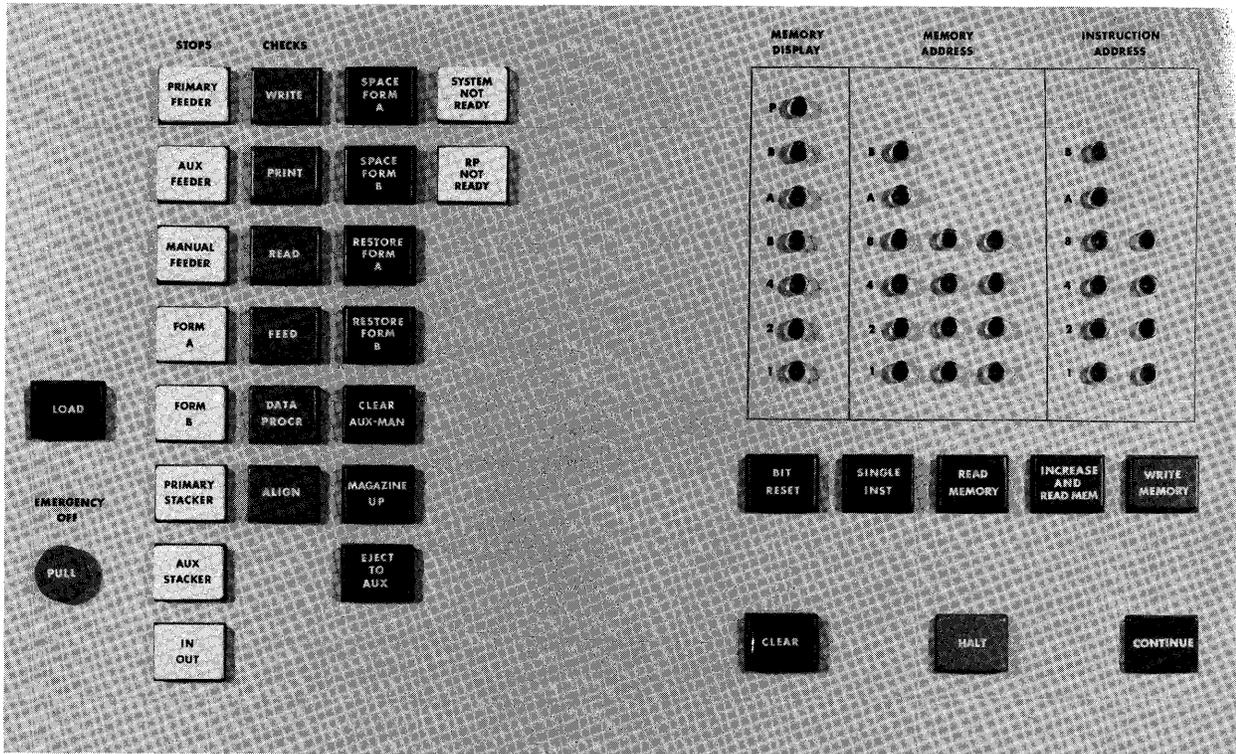


Figure 5-10. Console Control Panel

MEMORY DISPLAY REGISTER. A column of seven lights which displays one character at a time. The character displayed can be a character called from storage for viewing or changing or a character that was manually inserted in the register for transfer to storage. The fact that the lights can be set or cleared manually and individually allows changing and inserting characters in the register.

MEMORY ADDRESS REGISTER. Three columns of lights which display a storage address. The address displayed is either that of the location of a character from storage to be viewed in the Memory Display register or that of a location in which a character from the Memory Display register is to be stored. These lights can also be set and cleared manually and individually.

INSTRUCTION ADDRESS REGISTER. Two columns of lights which display the address of the next instruction to be executed. Since the third character of a machine language instruction address is always zero, only the first two digits need be displayed. These lights can also be set and cleared manually and individually.

The set of switches and indicators below the register displays are used to manipulate the information in the Memory Display and Memory Address registers and to control the mode of operation of the computer.

BIT RESET. This switch allows clearing of bits in the Memory Display, Memory Address, and Instruction Address registers. By depressing a switch corresponding to a light and the Bit Reset switch simultaneously, that light goes off.

SINGLE INSTRUCTION. Each time this switch is depressed the instruction starting at the address displayed in the instruction address register is executed.

READ MEMORY. When this switch is depressed, the character in the storage location indicated by the Memory Address register is displayed in the Memory Display register.

INCREASE AND READ MEMORY. Each time this switch is depressed, the contents of the Memory Address register are increased by one and the character stored in the new location whose address is displayed in the Memory Address register is displayed in the Memory Display register.

WRITE MEMORY. When this switch is depressed, the character displayed in the Memory Display register is stored in the location indexed in the Memory Address register.

CLEAR. When this switch is depressed, all indicators are cleared and all registers are set to zero.

LOAD. The Load switch initiates a Card Read instruction for reading in a series of cards and storing the data from these cards in successive memory locations starting with location 000 until memory is completely filled and the Instruction Address register is back to 000.

HALT. This is a combination switch and indicator. When the switch is depressed, the system halts after all operations in progress have been completed. The indicator goes on whenever the system halts.

CONTINUE. When this switch is depressed, the B 251 System continues under program control, starting with the instruction stored at the location whose address is in the Instruction Address register. This switch is also an indicator which is on whenever the system is in continuous operation.

On the rear of the Console control panel are six switches to control power and to raise and lower the magazine (Figure 5-11).

POWER ON. Turns power on for the system.

POWER OFF. Turns power off for the system.

TRANSPORT ON. Turns power on to all motors in the Record Processor.

TRANSPORT OFF. Turns power off to all motors in the Record Processor.

MAGAZINE UP. Raises the magazine in the primary feed station to the feed position.

MAGAZINE DOWN. Lowers the magazine in the primary feed station. The magazine also lowers itself automatically when it becomes empty.

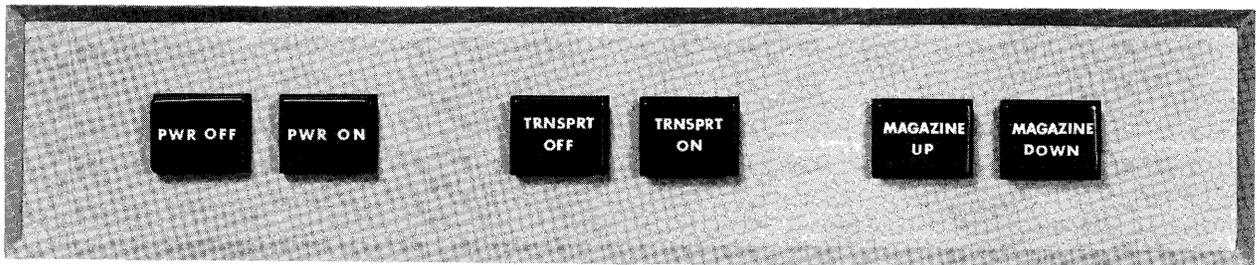


Figure 5-11. Power and Magazine Switches

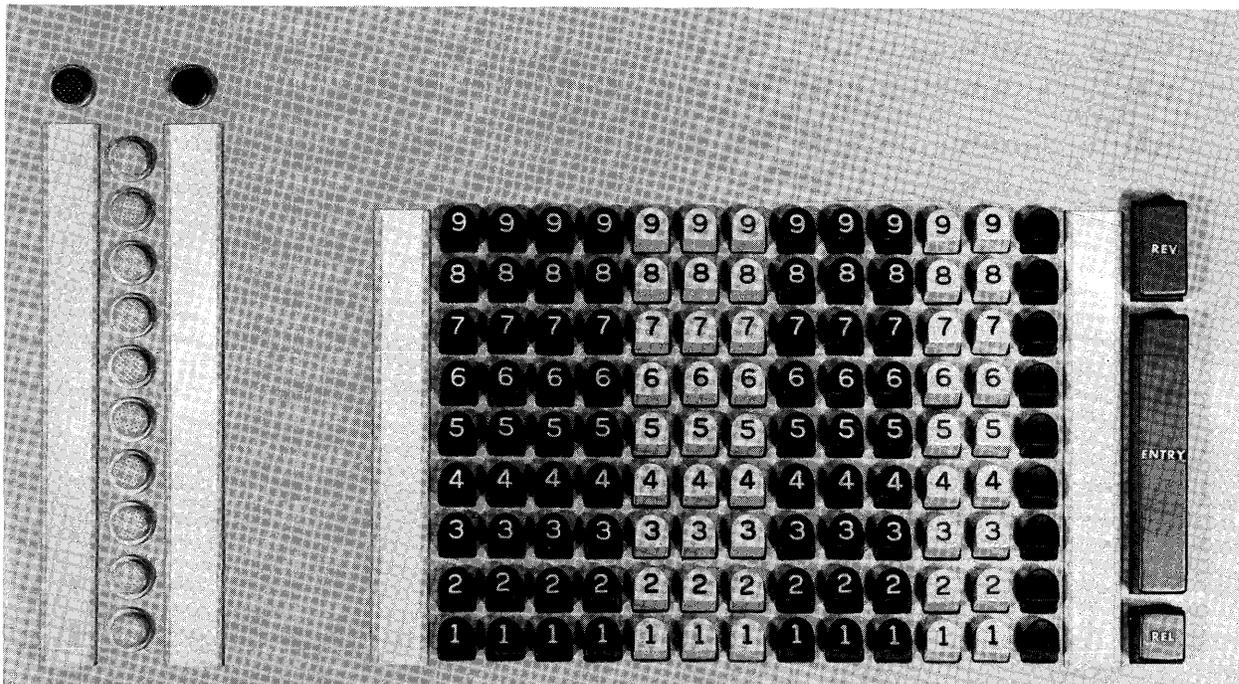


Figure 5-12. Console Keyboard

CONSOLE KEYBOARD

The Console Keyboard (Figure 5-12) provides a means of communication with the system by the operator. Through the keyboard the operator can externally control basic system functions.

The Keyboard consists of a bank of Numerical Data keys, a column of Special Data keys, Entry Control and Release keys, key identification strips, and indicators.

NUMERICAL DATA KEYS. Twelve columns of nine keys each are provided to introduce numerical data into the system. The keys in each column are numbered from one to nine, the number one being at the bottom. Zero is assumed when no key is depressed. After the Entry Control key is depressed, the Numerical Data keys are locked against further depression until released by a Data Processor instruction.

SPECIAL DATA KEYS. The Special Data keys form a single column of nine unlabeled keys to the right of the Numerical Data keys. This column of keys is also locked against further depressing after the Entry Control key is depressed. The numeric equivalent of this column will always be stored in the next character position to the right of the numeric data.

ENTRY CONTROL KEY. The Entry Control key is located to the right of the Special Data keys. This key does not latch down, but closes electrical contacts which allow the transfer of the keyboard con-

tents to the control unit upon execution of the Keyboard Read instruction.

REVERSE KEY. By means of the Reverse key a negative number can be stored from the Keyboard. When this key is depressed, the necessary minus sign bit will be stored over the least significant digit of the number being entered.

RELEASE KEY. A Release key is provided to release the entire keyboard in the event data has been indexed incorrectly. This key does not latch down and is inoperative once the Entry Control key is depressed.

KEY IDENTIFICATION STRIPS. Two key identification strips are furnished, one to the left of the Numerical Data keys and the other to the right of the Special Data keys. They can be easily changed by the operator.

INDICATORS. Located on the left section of the keyboard is a single column of indicators which will be on to indicate various conditions on the Halt and Keyboard Read instructions. On either side of this column of indicators is an identification strip. The strip on the left side will identify the various information to be indexed on the keyboard and an indicator will go on above this strip to show that the command being executed is a Keyboard Read. The strip to the right of the column of indicators is to identify various halt conditions and an indicator above this strip is provided to show that a Halt command has been executed.

SECTION 6

ASSEMBLER FOR THE B 251

VISIBLE RECORD COMPUTER SYSTEM

INTRODUCTION

An assembler is an automatic coding system, a method of making use of a computer to aid in the task of preparing programs. An assembler has its own language, just as a computer has its own language. When using an assembler, a program is coded in the language of the assembler. The assembler produces a program in computer language—the object program—from the program in assembler language. Coding a program in assembler language is much simpler and eliminates many of the problems involved in coding in the language of the computer. One helpful feature is the use of symbolic addressing when coding in assembler language. For example, when coding in machine language the discovery of a missing instruction three pages back has serious effects. When the missing instruction is inserted, all subsequent instructions must be moved up one location and thus all reference addresses must be changed. However, since an assembler allows the use of symbolic rather than absolute addressing, a missing instruction can be inserted anywhere with the proper symbolic address.

Another feature of an assembler is the ability to use the mnemonic operation codes of instructions. The task of assigning machine language characters for the instructions is left to the assembler.

The B 251 VRC assembler makes use of either the Burroughs 205 or 220 computer for assembly. The object program is in the language of the VRC System and is ready to be run on that system.

ASSEMBLER PROGRAMMING PROCEDURES

Basic instruction and constant coding for the assembler will be explained in the following discussion of the special coding form.

ASSEMBLER CODING FORM

Each line on the coding form (Figure 6-1) represents one constant or instruction and has been divided into 12 fields. In preparing the symbolic program it is

very important that all entries are made in the correct field and columns within the field.

IDENT. NO. (Col. 1-6). A six-column field for program identification. This field is located at the bottom of the coding form. A name or number can be inserted in any columns of this field for identification of the program being assembled.

OP. (Col. 7-9). A three-column, alphabetic field for the mnemonic op code of instruction entries; for example, ADD, MUL, CPA, etc. The letters CST must be entered here for the first of each group of constant entries or a working storage entry.

The letters HDG are entered in this field whenever specific heading information is to be listed with the program. The information to be listed is entered in the Constants and/or Remarks field(s). HDG must be entered in the Op field for each entry to be listed.

There are also two assembler pseudo operation codes (AAR and LNK) which are entered in this field. They will be discussed later.

M (Col. 10 and 11). A two-column alphanumeric field for designating the length of the A operand in all arithmetic instructions and the Mask instruction. This field is also used to designate form spacing control for the Print and Skip instructions. Only the numbers 00 through 12, inclusive, and certain alphabetic characters for Sorter and Record Processor instructions, may be entered in this field. (See Assembler Instruction List.)

N (Col. 12 and 13). A two-column, alphanumeric field for designating the length of the B operand in all arithmetic instructions. For the Compare instruction this field will indicate the number of characters to be compared. Form skipping requirements for the Print and Skip instructions should be indicated in this field. For the Transfer instruction this field indicates the number of characters to be transferred. (If the number of characters to be transferred exceeds 99, the hundreds digit is entered in the M

field.) The maximum number of characters that can be transferred is 120. With the exception of the Transfer instruction, only the numbers 00 through 12, inclusive, and certain alphabetic characters for Sorter and Record Processor instructions, may be entered in this field. (See Assembler Instruction List.)

The M and N fields may also be used for halt numbers to identify the cause of the halt.

A ADDRESS (Col. 14-17). A four-column, alphanumeric field used to designate the address of the A operand. The first two columns designate the page on which the field being addressed has been coded. The third column designates the reference symbol (alphabetic character in the reference column) of the field being addressed.

The fourth column designates the column heading of the character being addressed. The only legitimate characters that may be used in the fourth column are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, T, E, O (Op), M, N, A, B, C, J, K, and L.

For the purpose of character addressing of instruction entries, Op is considered as column 0, M as column 1, N as column 2, the three-character assembled (machine) A address as columns 3, 4, and 5 respectively, the three-character assembled B address as columns 6, 7, and 8 respectively, and the three-character assembled C address as columns 9, T, and E respectively. For example, if the A Address field were coded as 20B5, the 20 would identify all entries on page 20, the B would identify the entry on page 20 labeled B, and the 5 would identify the character in column 5 of the Constants field of that entry if a constant, or the low order digit of the assembled A address of that entry if an instruction.

The alphabetic column headings can also be used in the fourth column number of the A Address field. For example, if the A Address field were coded as 68AC, the character identified would be located on page 68, in the entry labeled A, and would be the high-order character of the C Address field.

The special column headings J and K may be used in the fourth position of the A Address field for the purpose of increasing an address. A J will tell the assembler to increase the assembled address by one 12-character storage field, and a K will tell the assembler to increase the assembled address by two 12-character storage fields. For example, if an A Address field coded 05A were assigned the location 300 during assembly, a later A Address field coded 05AJ would be assigned the location 310, and one coded 05AK would be assigned the location 320.

The addressing scheme described above is the one normally used. However, more advanced addressing techniques are available. They are described later in Special Addressing Techniques.

B ADDRESS (Col. 18-21). A four-column, alphanumeric field used to designate the address of the B operand. The same addressing scheme is used here as for the A address.

C ADDRESS (Col. 22-25). A four-column, alphanumeric field used to designate the address of the C operand. The same addressing scheme is used here as for the A address.

PAGE (Col. 26 and 27). A two-column, numeric field used to designate the page number of an entry. The page number with the reference symbol form an identity unique to any particular entry. The page number with the line number (see below) determine the sequencing of all entries. The page number need not be written for each entry on a page, but is required for at least the first entry. However, the page number will be punched into each entry card.

REFERENCE (Col. 28). A one-column, alphabetic field that is coded for every entry that is to be referred to by any other entry. Any alphabetic character may be used and no regard need be given to sequence. However, no character may be used more than once per page.

SIZE (Col. 29-31). A three-column, numeric field used to designate the length of a constant or a working storage entry. The maximum length of a constant entry is 012. The maximum length for a working storage entry is 999. When the Size field exceeds 012, the first 12 characters entered in the constant field will be assembled as a constant; the remaining positions specified will be reserved for a working storage area.

CONSTANTS (Col. 32-43). A 12-column, alphanumeric field for entering constants. A maximum of 12 characters per entry will be assembled. The actual number of characters assembled depends on the number entered in the Size field.

REMARKS (Col. 44-77). A 34-column, alphanumeric field. Twenty-eight of these columns may be used for remarks, regardless of the system used for assembly. The remaining 6 columns should be left blank: they are reserved for the assembler.

LINE (Col. 78-80). A three-column, numeric field used with the page number for input sequencing. The first two columns designate the line number on the page and the third column is reserved for inserts. This enables the programmer to patch in additional entries without having to recode all entries subsequent to the one being inserted.

CONTROL ENTRIES

There are three assembler directing operational codes provided with the assembler. These control codes allow specification of the starting machine location of the object program or regulation of the machine location sequence during assembly, and signaling the end of the symbolic entries to be assembled. Control entries never appear in storage but are listed with the program. When encountered during assembly, the job specified by a control entry is performed but the entry is not assembled with the object program.

Any specification or regulation of the machine location sequence during assembly is accomplished by setting or changing the setting of the location counter. The location counter is a three-character storage field which is set to 000 by the assembler at the beginning of assembly. The location counter is then counted up one each time a machine address is assigned. The initial setting and counting up of the location counter is automatic under control of the assembler unless otherwise specified by one of the control entries SLC or ALC.

SLC. This control entry allows setting the location counter to any desired value at any time during the assembly of the program. SLC is entered in the Op field and the value to which the location counter is to be set is placed in the second, third, and fourth columns of the A Address field. The Page and Line fields are used for page and line number. No other fields are required.

ALC. This control entry allows the programmer to adjust the location counter, that is, to regulate the units and tens positions of the location counter during assembly. It sometimes is desirable to have a group of instructions or constants assembled and stored starting at a specific storage location.

Op	M	N	A ADDRESS			B ADDRESS			C ADDRESS			PAGE	REF	CONSTAN																	
			PG	R	I	PG	R	I	PG	R	I			SIZE	0	1	2	3	4	5											
7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	
ALC						0															04										

To accomplish this, ALC is entered in the Op field. If both the units and tens positions are to be regulated, the desired settings are entered in the third and fourth columns of the A Address field. If only the units position is to be regulated, the desired setting is entered in the fourth column of the A Ad-

dress field and the third column must be left blank. The Page and Line fields are used for page and line number. No other fields are required.

END. This control entry flags the end of the symbolic entries to be assembled. It is required and is punched in the final card of the symbolic program deck.

END is entered in the Op field. The Page and Line fields are used for page and line number. No other fields are required.

SPECIAL ADDRESSING TECHNIQUES

The VRC assembler provides several short-cut addressing techniques in addition to the normal addressing method described above. These techniques are very useful in the special cases to which they pertain.

LITERALS. A literal is a particular piece of data which is referenced by a particular instruction. The purpose of a literal is to allow the referencing of the data without separately setting it up as a constant. A literal is entered on the same line as the instruction entry which refers to it. This is done by placing an asterisk (*) in the first column of the address field and placing the data to be referenced in the Constants field. Only one literal per entry, up to 12 characters in length, may be used.

For example, if it is desirable to test a character located at 18A5 for zero, this can be done under normal addressing by setting up a constant of zero at, say, 40C1. The instruction would then read:

Op	M	N	A ADDRESS			B ADDRESS			C ADDRESS			PAGE	REF	CONSTAN																	
			PG	R	I	PG	R	I	PG	R	I			SIZE	0	1	2	3	4	5											
7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	
C	P	A				1	1	8	A	5	4	0	C	1	X	X	X				68										

By using a literal the same thing could be accomplished without setting up a separate constant of zero. Using a literal the instruction would read:

Op	M	N	A ADDRESS			B ADDRESS			C ADDRESS			PAGE	REF	CONSTAN																	
			PG	R	I	PG	R	I	PG	R	I			SIZE	0	1	2	3	4	5											
7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	
C	P	A				1	1	8	A	5	*				X	X	X				68					0					

B 251 ASSEMBLER INSTRUCTION LIST

OPERATION	MNEMONIC	CODE	M	N	AAA	BBB	CCC
						ADDRESS OF MSD OF	
No-Operation	NOP						
Add	ADD	1	La(1-12)	Lb(1-12)	Augend	Addend	Sum
Subtract	SUB	2	La(1-12)	Lb(1-12)	Minuend	Subtrahend	Difference
Multiply	MUL	3	La(1-12)	Lb(1-12)	Multiplicand	Multiplier	Product
Divide	DIV	4	La(1-12)	Lb(1-12)	Dividend	Divisor	Quotient
Compare—							
Alphabetic	CPA	5		La&b(1-12)			Branch Equal
Zone	CPZ	5		La&b(1-12)			Branch Equal
Numeric	CPN	5		La&b(1-12)			Branch Equal
Branch—							
Conditional	BRC	6			Branch Low	Branch Equal	Branch High
Unconditional	BRU	6			Branch		
Transfer	TFR	7		No. of Char.	Source		Destination
Mask	MSK	8	La(1-12)		Source	Mask	Destination
Halt	HLT	9	Identification	Identification			
Card Read	CRD					End of File	Destination
Sorter Read—							
Demand	SRD			D-Demand	Can't Read	End of File	
Flow	SRF			E-End of Document	Can't Read	End of File	
				T-Through Transit No.	Can't Read	End of File	
Control Sorter	CTL	C	P-Pocket Select Only	0-9, X, Y, R			
			D-Demand Feed	0-9, X, Y, R			
			S-Stop Flow	0-9, X, Y, R			
			C-Start or Cont. Flow	0-9, X, Y, R			
Record Processor							
Read	RPR	G	P-Primary M-Manual A-Auxiliary V-Optical K-Keyboard		Read Error		Destination
Write and Eject	RPW	G	W-Write and Eject	P-Primary A-Auxiliary	Source		Write Error
Eject Only	RPW	G	E-Eject	P-Primary A-Auxiliary			
Print	PRV	H	E-Eject		Source	Form B-Last Line Branch	Print Error
				0-Space Suppress			
				1-Space Form A			
				2-Space Form B			
				3-Space Forms A & B			
				4-Space Record			
				5-Space Form A & Record			
				6-Space Form B & Record			
				7-Space All Three			

B 251 ASSEMBLER INSTRUCTION LIST—Continued

OPERATION	MNEMONIC CODE	M	N	AAA	BBB	CCC
					ADDRESS OF MSD OF	
Skip	SKR	I	When M = 2; Otherwise Blank When M = 4			
			1-Space Form A	Line Find Address	Form B-Last Line Branch	
		1-Restore Forms	2-Space Form B			
		2-Single Space Record	3-Space Forms A & B			
			4-Space Record			
		4-Align Record	5-Space Form A & Record			
			6-Space Form B & Record			
			7-Space All Three			
La—Length of AAA field						
Lb—Length of BBB field						
La & b—Length of AAA and BBB fields (not total)						
Heading	HDG					Constants/ Remarks
Constants	CST					
Set Location Counter	SLC			Machine Language Location Setting		
Set Tens and/or Units position of Location Counter	ALC					
End of Symbolic Program	END					
Address Arithmetic	AAR			Address of Data to be Modified	Address of Modifier	Address Where Modified Data is to be Stored
Linkage	LNK			Address of First Subroutine Instruction		

APPENDIX A

WORD FORMATTING OF MICR INFORMATION IN STORAGE

Documents processed by the B 101 Sorter-Reader have magnetically encoded characters which, on passing the read head, are either transferred to the sorter buffer or directly to core storage, depending on the sorter instruction employed. In either case, a minimum of 84 characters of information are stored in seven storage fields.

Each field on the document is defined by symbols. Reading from right to left is the amount (start) symbol, the amount field, a second amount symbol, the transaction code field, the on-us symbol, the account number field, etc.

The first amount symbol initiates reading and is stored in the least significant digit position of the input area (CCC + 83). The 10 digits of the amount field are stored starting in the second least significant digit position. The second amount symbol signals the end of the amount field and is stored in the most significant digit position to complete the first 12-character field of storage.

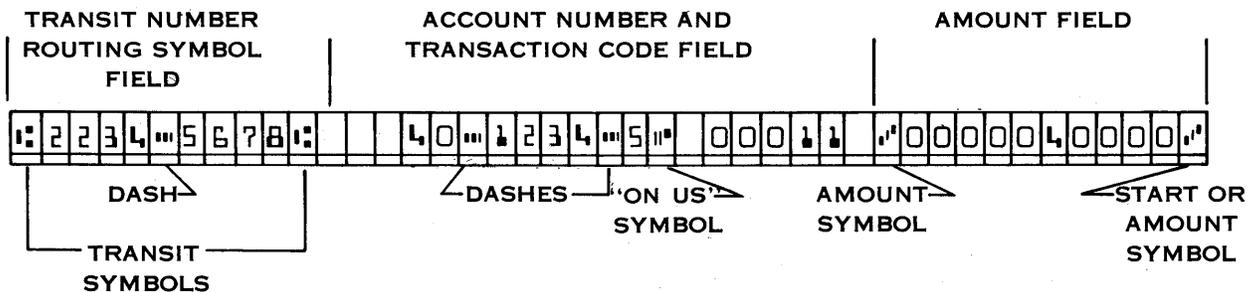
Storage of data continues starting in the least significant digit position of the next storage field to the left until the next symbol is sensed, at which time that symbol is stored in the next position and the remainder, if any, of that field is blanked out. Storage continues in this manner until either the ending transit symbol is sensed and stored or all the information on the document has been stored, depending on the Sorter Read instruction used. If less than seven fields of information are encoded on a document or through the Sorter Read instruction, less than seven fields are read, the remaining 12-character fields are blanked out.

The order of the symbols in the following examples adheres to the ABA standards. Where other item formats are used, such as outside the United States, no machine changes are required. The symbols will be recognized as programmed.

Example :

O	M	N	AAA	BBB	CCC
#	0	4	400	300	144

Read Instruction Employed



Pre-Encoded MICR Document

APPENDIX B

B 251 COMMAND LIST

OPERATION	MNEMONIC	CODE	M	N	AAA	BBB	CCC
No-Operation	NOP						
Add	ADD	1	La(0-11)	La(0-11)	Augend	Addend	Sum
Subtract	SUB	2	La(0-11)	La(0-11)	Minuend	Subtrahend	Difference
Multiply	MUL	3	La(0-11)	La(0-11)	Multiplicand	Multiplier	Product
Divide	DIV	4	La(0-11)	La(0-11)	Dividend	Divisor	Quotient
Compare—							
Alphabetic	CPA	5	0	La&b(0-11)			Branch Equal
Zone	CPZ	5	1	La&b(0-11)			Branch Equal
Numeric	CPN	5	2	La&b(0-11)			Branch Equal
Branch—							
Conditional	BRC	6	0		Branch Low or < 0	Branch Equal or 0	Branch High or > 0
Unconditional	BRU	6	1		Branch		
Transfer	TFR	7	Words (0-9)	Charact. (0-11)	Source		Destination
Mask	MSK	8	La(0-11)		Source	Mask	Destination
Halt	HLT	9		Identification			
Card Read	CRD	#		1		End of File	Destination
Sorter Read							
Demand	SRD	#		6	Can't Read	End of File	Destination
Flow	SRF	#		4-All Fields 5-Thru Transit	Can't Read	End of File	Destination
Control Sorter	CTR	C	0-Pocket Select Only 2-Demand Feed 4-Stop Flow 6-Continue Flow	Pocket 0-11, 15			
Record Processor							
Read	RPR	G	1-Primary 2-Manual 3-Auxiliary 4-Optical 5-Keyboards		Read Error Read Error Read Error		Destination Destination Destination Destination
Write and Eject	RPW	G	8	0-Primary 1-Auxiliary	Source Source		Write Error Write Error
Eject	RPW	G	9	0-Primary 1-Auxiliary			
Print	PRV	H		0-Suppress 1-Space Form A 2-Space Form B 3-Space A & B 4-Space Record 5-Record & A 6-Record & B 7-Space All	Source	Last Line Branch	Print Error
Skip	SKR	I	0-Suppress 1-Restore Forms 2-Single Space 3-Restore Forms & Space Record 4-Align	When M equals 2 1-Form A 2-Form B 3-Forms A & B 4-Record 5-Record & A 6-Record & B 7-All Three	When M equals 4 Line Find	Last Line Branch	

APPENDIX C

VRC CODE REFERENCE TABLE (COLLATING SEQUENCE)

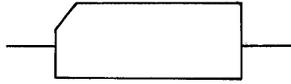
PUNCHED CARD CODE	026 SYMBOL	B 251 CORE STORAGE CODE							B 251 SYMBOL	B 251 ADDRESSING SEC. FIELD CHAR.	B 251 PRINT BAR POSITION	MICR SYMBOL	B 101 POCKET SELECT CODE	B 251 ACCT. RECORD CODE	COMMENTS
		P	B	A	8	4	2	1							
"Blank"	"Space"	1	1	1					"Blank"	30	Blank				Blank
12 8,3	.			1	1			1	.						Decimal
12 8,4	□	1		1	1			1	□						Lozenge
12 8,5	&			1	1	1			&						Ampersand
12 8,6		1		1	1	1		1							
12 8,7	≠			1	1	1	1	1							Group Mark
11 8,3	\$		1		1			1	\$					-10	Dollar Sign
11 8,4	*	1	1		1			1	*					-11	Asterisk
11 8,5	-		1		1			1	-					-12	Minus Sign
11 8,6		1	1		1	1		1						-13	
11 8,7			1		1	1	1	1						-14	
0 8,1	/		1		1			1	/	31				-15	Slash
0 8,3	,	1	1	1	1			1	,						Comma
0 8,4	%		1	1	1			1	%						Per cent Sign
0 8,5			1	1	1	1		1							
0 8,6			1	1	1	1	1	1							
0 8,7		1	1	1	1	1	1	1							
8,3	#	1			1			1	#	10	10	amount, I ¹	x	10	Number Sign
8,4	@				1			1	@	11	11	transit I ²	y	11	At Sign
8,5					1	1		1				on-us I ³		13	
8,6					1	1	1	1						14	
8,7	≠	1			1	1	1	1				reject		15	Tape Mark
12 0	o			1					o	10					
12 1	A	1		1				1	A	11					
12 2	B	1		1				1	B	12					
12 3	C			1				1	C	13					
12 4	D	1		1		1			D	14					
12 5	E			1		1		1	E	15					
12 6	F			1		1	1		F	16					
12 7	G	1		1		1	1	1	G	17					
12 8	H	1		1	1				H	18					
12 9	I			1	1			1	I	19					
11 0	o		1						o	20					-0
11 1	J	1	1					1	J	21					-1
11 2	K	1	1					1	K	22					-2
11 3	L	1	1					1	L	23					-3
11 4	M	1	1			1			M	24					-4
11 5	N		1			1		1	N	25					-5
11 6	O		1			1	1		O	26					-6
11 7	P	1	1			1	1	1	P	27					-7
11 8	Q	1	1			1			Q	28					-8
11 9	R	1	1			1		1	R	29					-9
0 8,2	≠	1	1	1	1	1									Record Mark
0 8,2	S		1	1				1	S	32					
0 8,3	T	1	1	1				1	T	33					
0 8,4	U		1	1		1			U	34					
0 8,5	V	1	1	1		1		1	V	35					
0 8,6	W	1	1	1		1	1		W	36					
0 8,7	X		1	1		1	1	1	X	37					
0 8,8	Y		1	1	1				Y	38					
0 8,9	Z	1	1	1	1			1	Z	39					
0 0	0	1							0	0	0	0	0	0	
1 1	1							1	1	1	1	1	1	1	
2 2	2							1	2	2	2	2	2	2	
3 3	3	1						1	3	3	3	3	3	3	
4 4	4					1			4	4	4	4	4	4	
5 5	5	1				1		1	5	5	5	5	5	5	
6 6	6	1				1	1		6	6	6	6	6	6	
7 7	7					1	1	1	7	7	7	7	7	7	
8 8	8					1			8	8	8	8	8	8	
9 9	9	1				1		1	9	9	9	9	9	9	
All Others		1				1	1								12

APPENDIX D

FLOW CHART STANDARD SYMBOLS

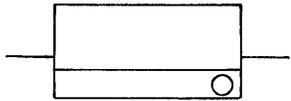
INPUT-OUTPUT

1. CARD READ



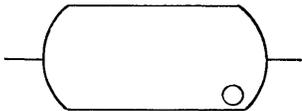
This symbol is used to designate the reading of one punched card.

2. SORTER READ



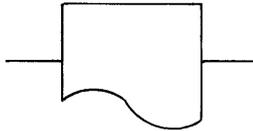
This symbol is used to indicate the reading of one MICR encoded document. In the small circle to the right is placed a "D" for Demand or an "F" for Flow to designate mode. This symbol is also used for the Control Sorter command with no circle.

3. RECORD PROCESSOR



This symbol is used to indicate the reading, writing, and/or ejecting of one account record. The small circle to the right will contain a letter indicating the source or destination of the record, such as: P—Primary, M—Manual, A—Auxiliary, V—Optical, or K—Keyboard.

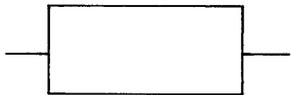
4. RECORD PROCESSOR PRINTER



This symbol is used to designate the control of the records and forms in the printing station. It will indicate printing of one line of information and single spacing of the record and forms and skipping over folds for the forms.

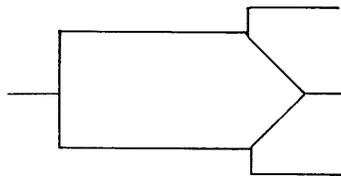
LOGIC

1. OPERATIONS



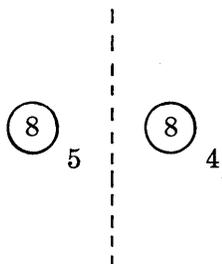
A symbol that is used to describe a nondecision type of operation such as "reduce account balance" or "clear printer output area." It is used for Add, Subtract, Multiply, Divide, Transfer, and Mask.

2. DECISIONS



A one-entry, two or three exit symbol. It is used to show that two or three-way conditional branch is possible due to a comparison or arithmetic function. A comparison of two values will result in one number being less than, equal to, or greater than the other number. The result of an arithmetic function will be minus, zero, or plus.

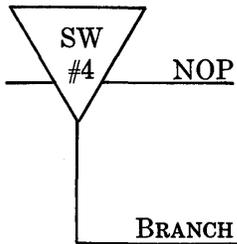
3. CONNECTORS



At such time as either space, logic, or convention dictates that a new page be started, or reference made to a previous page, this symbol is used as shown. The number in the circle identifies the connector. For exit connector, the subscript number identifies the page on which the entry connector is to be found. For entry connectors, the subscript number specifies the location (page) of the exit connector. Subscript zero means the exit and entry connectors are on the same page. Common practice calls for sequential numbering of connectors.

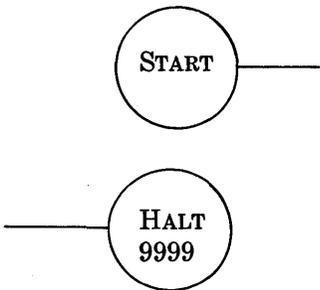
APPENDIX D—Continued

4. SWITCHES



This is a one-entry, two-exit symbol to designate a two-way switch. It is used to show manipulation of the Branch and No-operation commands.

5. START OR HALT



This symbol is used to signify the start of a program or an unconditional program Halt.

APPENDIX E

TIMING CHART

INTERNAL OPERATIONS

INSTRUCTION NAME	INSTRUCTION CODE	TIME OR FORMULAE
NO-OPERATION	BLANK	40 μ s
ADD	1	$[4+4(M+N)+6L]10.5\mu s$
SUBTRACT	2	$[4+4(M+N)+6L]10.5\mu s$
MULTIPLY	3	$[3+6N+21MN]10.5\mu s$
DIVIDE	4	$[2+(M-N)(70N+63)]10.5\mu s$
COMPARE	5	$[5+10N]10.5\mu s$ add 30 μ s for equal branch condition
BRANCH	6	75 μ s
TRANSFER	7	$[4+8(12M+N)]10.5\mu s$
MASK	8	$[4+12M+8I]10.5\mu s$
HALT	9	300ms

M—NUMBER OF CHARACTERS IN FIELD A

N—NUMBER OF CHARACTERS IN FIELD B

L—NUMBER OF CHARACTERS IN FIELD A OR FIELD B, WHICHEVER IS GREATER

I—NUMBER OF CHARACTERS INSERTED BY MASK

ms—MILLISECONDS

μ s—MICROSECONDS

INPUT-OUTPUT OPERATIONS

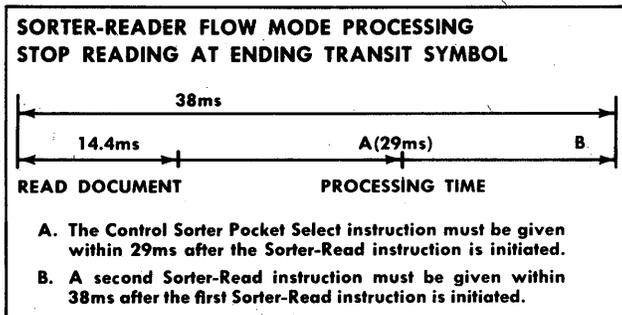
Card Reader

CARD READ (#)	2ms	Transfer 80 characters from card buffer to storage. Upon completion of the transfer to storage the next card will be read into the buffer in 600ms.
---------------	-----	---

Sorter-Reader

SORTER READ (#)		
DEMAND MODE	2ms	Transfer 84 characters from Sorter-Reader Buffer to storage.
FLOW MODE	14.4ms	Read one document directly to storage, stop reading at the ending transit symbol.
	23.8ms	Read one maximum size (9.5 inches) document directly to storage. Stop reading at end of document.
CONTROL SORTER (C)	120 μ s	The control sorter instruction to pocket select in flow mode must be given within 29ms after the start of the sorter read flow mode instruction. Following a control sorter instruction to pocket select and demand feed one document, the document will be read into the buffer in 150ms.

APPENDIX E—Continued



Record Processor

Record Processor Read (G)

Read Ledger—Primary Feed	282ms
Read Ledger—Auxiliary Feed	295ms
Read Ledger—Manual Feed	295ms
Read Bar Code—	
Auxiliary Feed	270ms
Read Keyboard	1.23ms

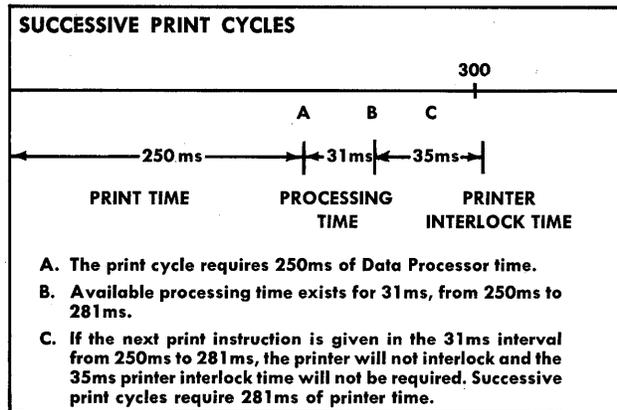
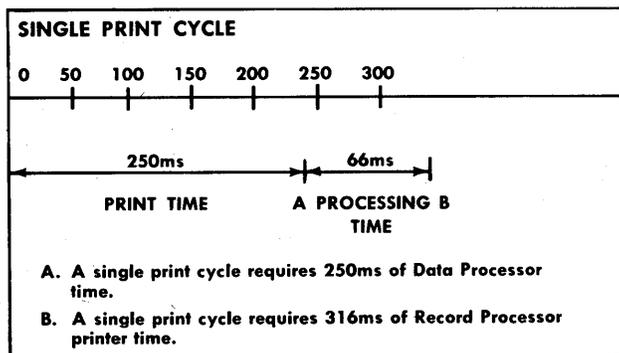
Record Processor Write and Eject (G)

Write and Eject—to primary or auxiliary stacker	
from line 0	630ms approx.
from line 24	550ms approx.
from line 47	474ms approx.

Eject only—to primary stacker 50 μ s.

Eject only—to auxiliary stacker 300ms.

Print on Record Processor(H) 250ms.



Spacing using the print on Record Processor instruction requires 40ms. The 40ms space time is included in the 250ms print time.

Skip Record Processor (I) 120 μ s Data Processor Time
Single line space requires 35ms for each line. The 35ms is available processing time.

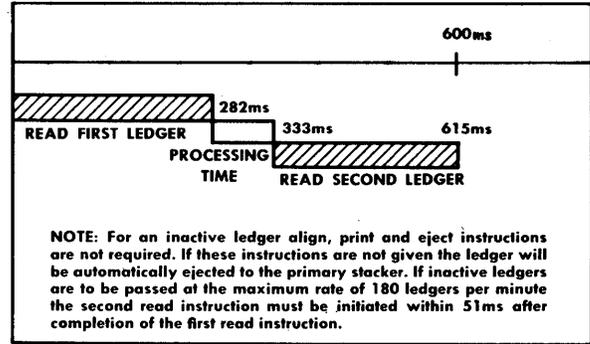
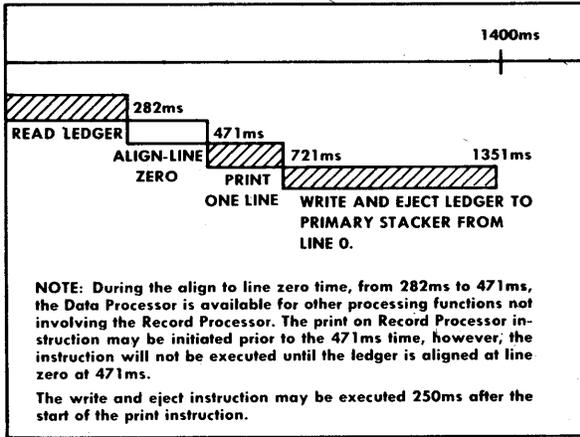
Align—From the end of the Record Processor Read instruction to align to line zero requires 189ms; for each additional line add 3.33ms. The align time is available processing time.

Align line zero	189ms approximately
Align line 24	269ms approximately
Align line 47	345ms approximately

Several examples of timing including ledger reading, aligning, printing, and writing are given below. Shaded areas indicate the Data Processor is not available for other processing; clear areas indicate the Data Processor is available for other processing which does not involve the Record Processor.

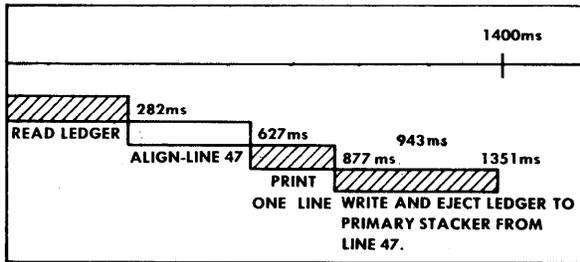
A—An active account to be aligned at line 0. Read a ledger from the primary feeder, align to line zero, print one line and write and eject the ledger to the primary stacker.

APPENDIX E—Continued



B—Active Account to be aligned at line 47.

Read a ledger from the primary feeder, align to line 47, print one line and write and eject the ledger to the primary stacker.



C—Inactive Account.

Read a ledger from the primary feeder, allow the ledger to automatically eject to the primary stacker, and read the next ledger from the primary feeder.



Burroughs Corporation

DETROIT 32, MICHIGAN

IN CANADA, BURROUGHS ADDING MACHINE OF CANADA, LTD., TORONTO, ONTARIO