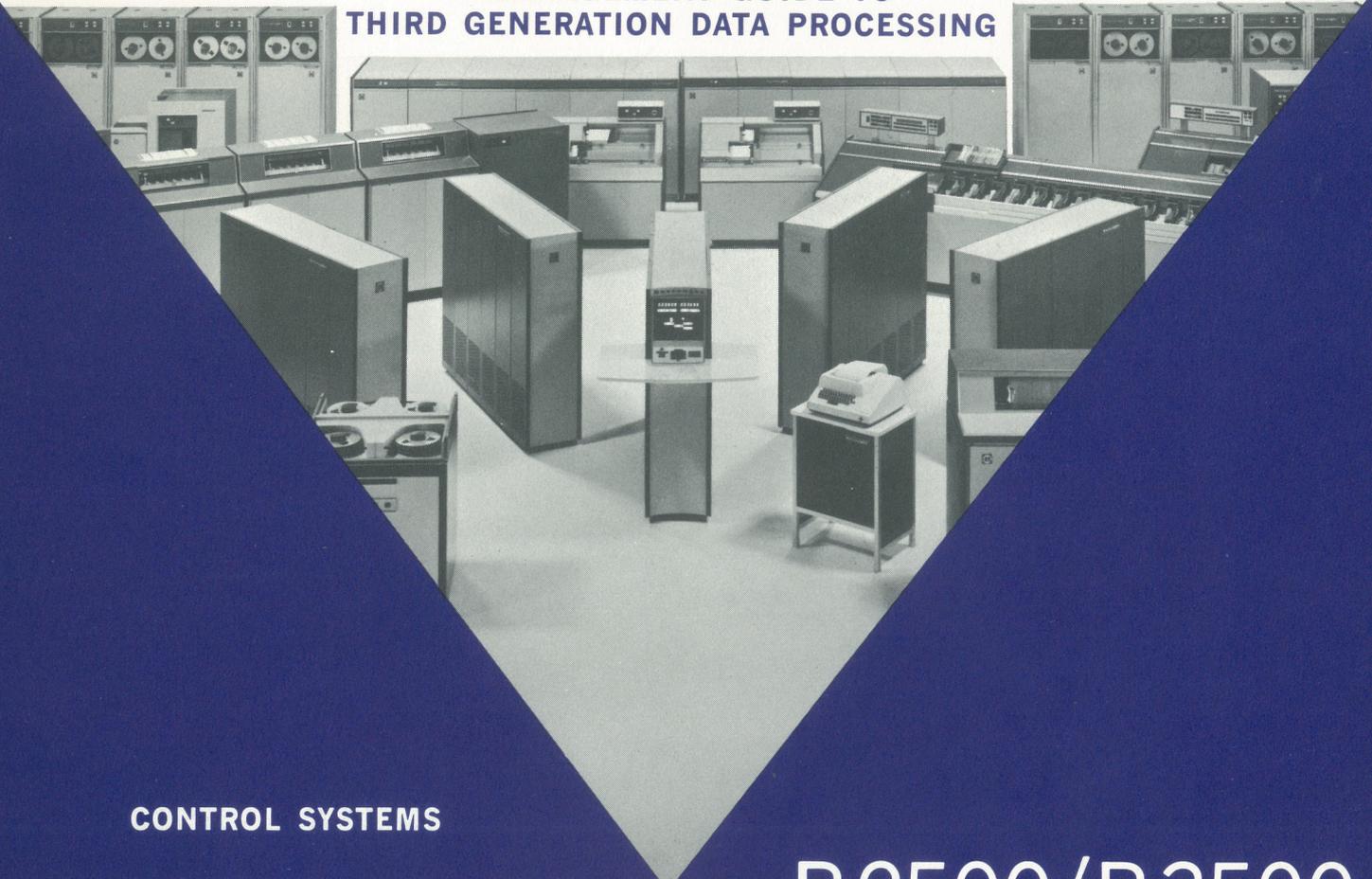


A MANAGEMENT GUIDE TO  
THIRD GENERATION DATA PROCESSING



CONTROL SYSTEMS

HARDWARE/SOFTWARE INTEGRATION

MULTIPROCESSING

B2500/B3500

SOFTWARE  
SYSTEMS





Today's EDP Manager often finds he can no longer meet increased demand on his department merely by installing more equipment or a faster system. This hardware approach often provides only a temporary solution to his workload problem. It fails to provide the responsiveness to unscheduled demands, applicational changes, and new applications required in most installations. Even excellent hardware is not enough to solve these problems, or to provide the high equipment utilization required for profitable EDP operation.

Software/hardware integration makes the difference. Burroughs B 2500 and B 3500 systems are a blend of advanced electronics and excellent systems programming. They are designed to allow fast response to change in workload or application. Multiprocessing facilities allow these machines to serve many programs simultaneously and to accept unscheduled jobs as easily as more routine processing. And their advanced multiprocessing techniques significantly improve equipment utilization.

Multiprocessing and full operating system control are provided by the B 2500 and B 3500 systems' Master Control Program. Installations that do not require multiprocessing may enjoy many elements of control program operation through use of a Basic Control Program.

Each B 2500 and B 3500 user receives a complete software package which includes, in addition to the control program, COBOL, Fortran and the Assembler Programming System to meet his major programming requirements. Generative software is used to meet day-to-day report, sort, and utility program needs.

This brochure describes some of the initial software offerings available with the Burroughs B 2500 and B 3500 systems. The reader should bear in mind that both the Basic and Advanced levels of B 2500 and B 3500 software are thoroughly supported by many hardware design features, and are, in fact, full partners with the equipment in delivering an unusually high performance-to-cost ratio.

## CONTENTS

Operating Systems .....	2
COBOL .....	8
Fortran .....	11
Assembler Programming Systems .....	12
Report Program Generators .....	13
Sort Program Generators .....	14
Media Conversion Program Generators ..	16

# OPERATING SYSTEMS

In early 1960, Burroughs Corporation adopted a new computer design philosophy. The corporation gathered its best software people and trained them in the basics of hardware design. At the same time, it steeped its best hardware designers in the art and science of software development. When crosstraining was completed, these people were molded into a design group with the task of producing a more effective computer system. The resulting Burroughs B 5000, and its successor, the B 5500, revolutionized the computer industry.

Today, Burroughs B 5500 users are acknowledged as one of the most satisfied groups of computer users in the world. Multiprocessing and automatic program and input/output scheduling, memory allocation, system logging, and library maintenance have proved

## MULTIPROCESSING

One of the major advantages of comprehensive operating system control is the ability to simultaneously process several programs. The Master Control Program (MCP) supplied with B 2500 and B 3500 disk file system configurations directs a group of programs through the system according to scheduling priority, automatically allocating memory areas for each, assigning peripheral equipment to meet I/O requirements, and tending to all processing details (commonly referred to as "housekeeping"). Independently written programs may be entered into the system in any combination of B 2500 and B 3500 languages and in any sequence. The MCP continues to assign hardware components to meet the requirements of incoming programs until the entire system is in operation. As soon as a program is terminated, the MCP allows another program or group of programs to refill the system. In this way, the MCP keeps the system at peak utilization as long as there is work to be done.

successful in B 5500 customer installations over a period of years. Today, this type of automatic operation has been generally accepted as a design requirement for a full third-generation computer system.

The same principles used to develop the revolutionary B 5500 have been applied—with the backing of experience—to the other Burroughs 500 Systems. The world's largest commercial computer system, the B 8500, operates in a multiprocessing mode under full automatic software control, as does the new large scale, thin-film B 6500. Late in 1966, Burroughs Corporation began demonstrating this full operating system control on the B 2500 and B 3500 systems. These systems now bring benefits formerly available only in larger systems into the economic reach of nearly all computer users.

A job may be entered into the system to meet an unexpected demand, given precedence, and, if desired, allowed to displace lower priority work currently being processed. As soon as the rush job is completed, regular work is resumed at the exact point where it was interrupted.

Some of the major MCP functions which have made multiprocessing so successful are explained below.

## PROGRAM LOADING AND SCHEDULING

Programs are accepted into the system from cards, paper tape, magnetic tape, or disk file, and may be in machine, compiler, or assembler language. The MCP automatically loads each program into disk file storage. When the program is requested, the MCP analyzes it for priority and equipment requirements, updates various tables for scheduling, and loads it from the library. The load routines written

for conventional systems are eliminated by these automatic MCP operations.

The MCP queues each program by priority and lists its equipment requirements as it is loaded. This program reference table is matched with a table reflecting the current status of each hardware component in the system and the highest priority program, with requirements matching the system's available resources, is initiated. This matching procedure is repeated each time there is a change in the system environment. The speed, detail and accuracy required for effective scheduling in a multiprocessing environment would be impossible to handle manually, yet taxes little of the system's own power. It results in much higher system productivity by assuring more continuous use of each hardware component.

## I/O SCHEDULING

The MCP maintains tables indicating the status of each I/O device as well as each I/O request being processed or awaiting an opportunity to be processed. Each time an I/O operation is initiated or terminated, the MCP compares the two lists and attempts to initiate all possible I/O operations. As a result, the system's I/O devices are kept fully active as long as there is work for them to do.

## FILE HANDLING

On an MCP controlled B 2500 or B 3500 system, the operator puts tape files for scheduled programs on any convenient tape unit. When the unit is activated, the MCP reads the file label, associates it with the unit, and thus is able to identify it when the program is initiated. The MCP also automatically assigns scratch tapes as needed. If a needed file or scratch tape is not mounted when the program is started, the MCP will advise the operator and start the program as soon as the requirements are met. Files from random access disk storage are allocated space automatically by the MCP and accessed by symbolic relative addresses in the object program. When a program is terminated, the MCP automatically closes all files as part of its

routine housekeeping function.

Files may consist of fixed or variable length records arranged in fixed or variable length blocks within core memory size limitations. The MCP automatically blocks and unblocks records according to specifications of the program.

## PRINTER BACK-UP

In a multiprocessing environment it is quite possible for several programs to require printer output simultaneously. The MCP handles this situation by diverting printer output to a scratch tape while the printer is busy and then reading the tape-captured data back to the printer when it is free. This allows processing to continue on programs which would otherwise have been suspended awaiting printer availability.

## MEMORY ALLOCATION

At load time the MCP automatically reserves and loads all disk file and core memory space necessary for each program, its associated data, and its I/O buffers. The MCP takes advantage of base relative addressing to "float" programs and data in memory to optimize the use of available core space. When a program is terminated, the MCP immediately carries out all housekeeping necessary to make memory space available to the next program and may relocate programs currently in memory to prevent "checkerboarding" and further maximize space availability.

## AUTOMATIC OVERLAY

Compiler and Assembler programs may be segmented to allow most of the program to remain in disk storage during execution until needed in core memory. The MCP automatically handles overlay routines, shuttling program segments from disk to core memory. This technique allows extremely large programs to be run on systems with modest core memory size, reduces core storage requirements for multiprocessing, and lowers overall system cost without sacrificing performance.

## OTHER MCP FUNCTIONS

In addition to the many MCP functions which contribute directly to the B 2500 and B 3500 systems' multiprocessing power, the MCP carries out a number of automatic functions which not only aid processing but also simplify system operation and computer department management.

### OPERATOR-SYSTEM COMMUNICATIONS

Most communications are from the MCP to the operator. The MCP informs the operator whenever it initiates or terminates a program. It advises him when a file is missing or a new scratch tape must be mounted and it warns of error conditions. The operator at any time may request a listing of all programs in the library or in process, and can start or suspend programs, change priorities, purge files, rewind tapes, and perform other operations through commands to the MCP.

### LOGGING

Many difficult cost accounting problems are completely solved through use of the MCP-generated log. On operator command, the MCP interrogates its files and prints out a

distribution of processor and peripheral time used by each program, plus a proration of system overhead time incurred. This log provides an excellent base for job costing and, as a complete and accurate picture of system utilization, greatly enhances management control over EDP operations.

### LIBRARY MAINTENANCE

The MCP allocates and maintains the system's library. It automatically assigns data and programs to the library, fetches them as required, and keeps track of their usage. The library may be maintained on magnetic tape or disk file or both and may be in source language, machine language, or both, thus affording complete flexibility at the user's option.

### SYSTEM CLOCK

The MCP maintains a 24-hour clock. It is used to supply time to object programs, to record timings in the log, and to trigger environment checking at regular intervals. Each time the MCP polls the equipment environment, it immediately takes action to make use of newly activated devices.

## BASIC CONTROL PROGRAM

Installations that do not require a full operating system or multiprocessing capabilities may use the Basic Control Program to obtain basic elements of operating system control. The BCP consists of a group of control programs which relieve the programmer of virtually all housekeeping associated with memory allocation, I/O initiate and terminate routines, and program loading and termination procedures.

At load time, the BCP determines the program's memory requirements and the relative address of its first executable instruction. It loads the program, carries out initialization routines such as setting base and limit registers, and then begins processing.

When the BCP receives an I/O request from the user program, it checks the avail-

ability of the required I/O channel. It will hold the request until the channel is free, initiate the I/O operation and return control to the user program. The BCP also checks for unresolved, non-normal conditions resulting from previous operations and prevents further use of the unit until these conditions are resolved. When an I/O operation is complete, the BCP makes the unit available for further use.

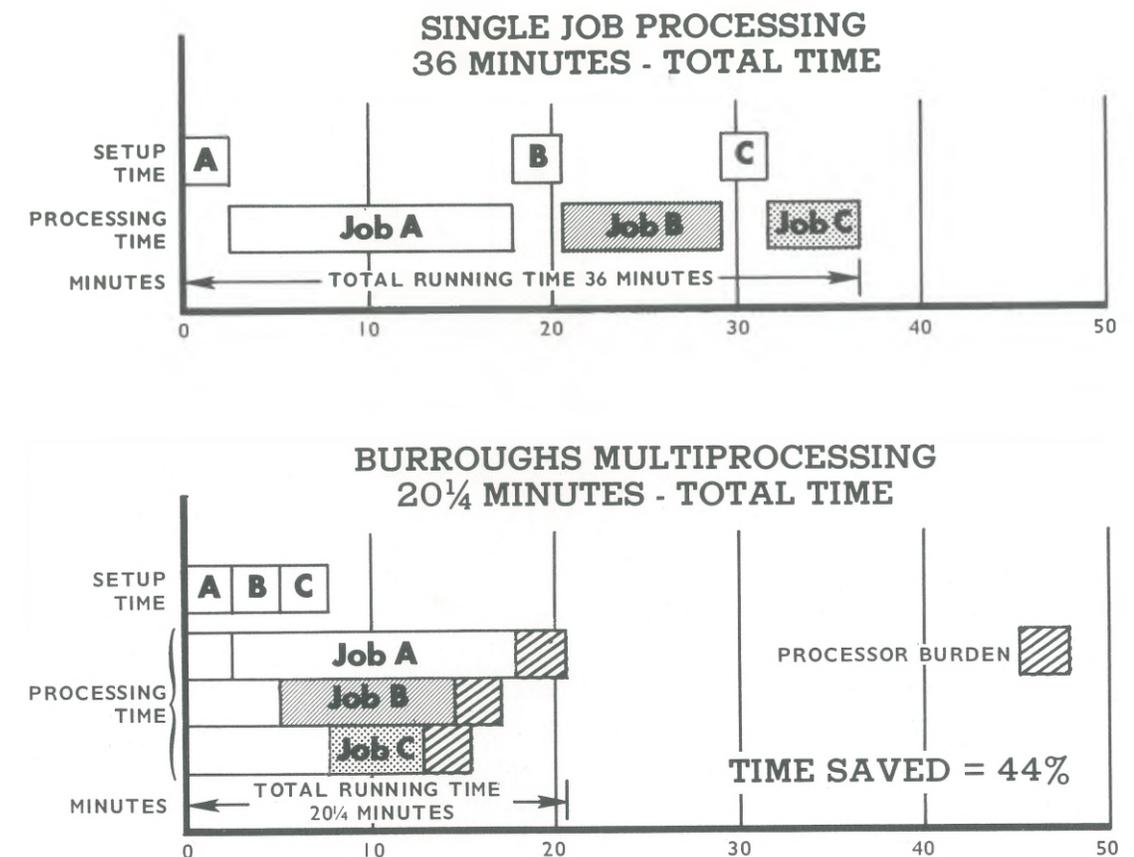
At the end of a program, the BCP will, at the operator's option, halt the system, load the next job and start processing, or load only the parameters for the next program. Use of the BCP greatly simplifies many programming and operating procedures and increases system efficiency.

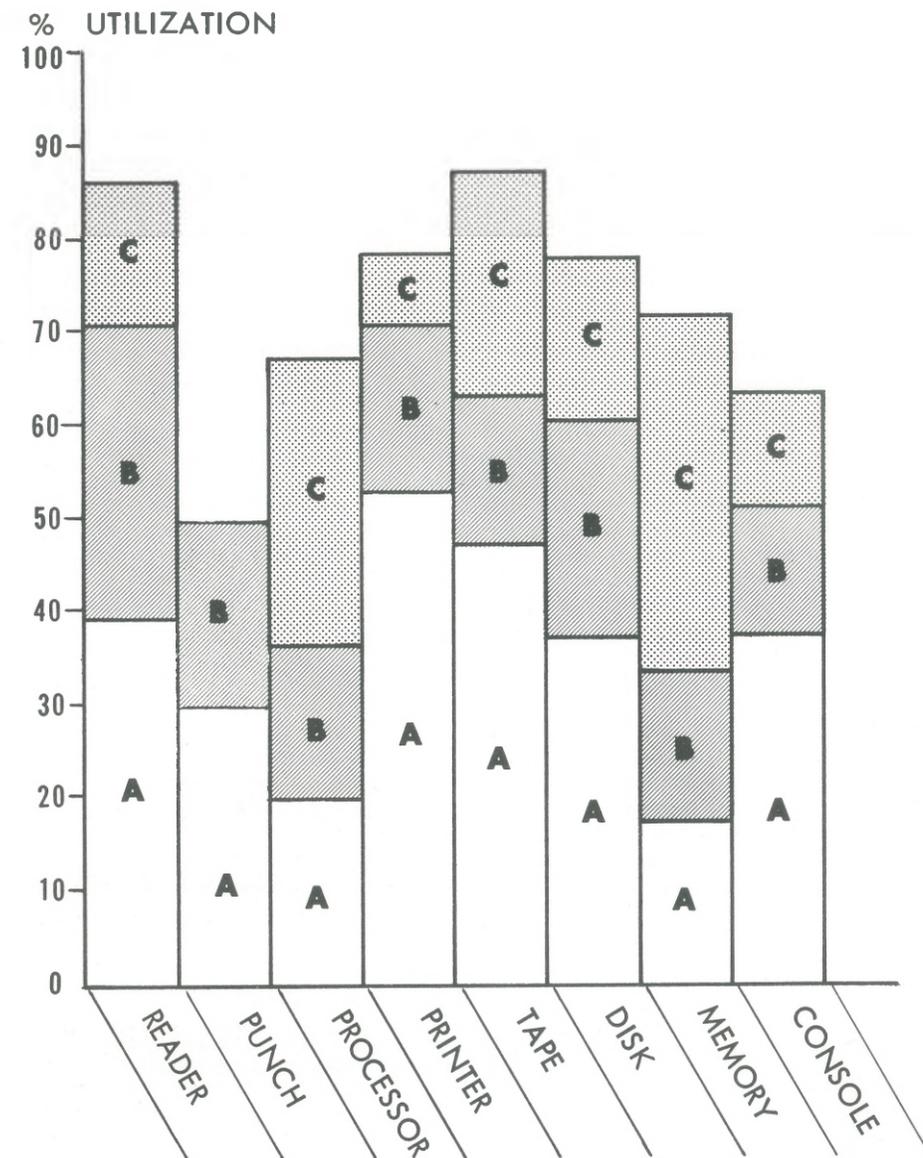
## MAJOR ADVANTAGES OF OPERATING SYSTEM CONTROL

### SYSTEM UTILIZATION

Multiprocessing under control of the MCP adds a new dimension to system utilization. In conventional systems, the configuration is designed to meet the requirements of the most demanding job. When less demanding programs are run, much of the equipment is idle. On the multiprocessing B 2500 and B 3500 systems, idle equipment is detected by the MCP and put to work. Programs are

continually called into the system until either the system's resources or the list of waiting programs has been exhausted. The chart below contrasts multiprocessing and conventional one-at-a-time processing. The time savings in this instance is 44 percent. In the graph on page 6 you may follow each of these jobs through the system and see just how much they use each component.





In this chart, Job A is a typical data processing run, with cards updating tape records, punching summary information, and printing reports. Job B is a file maintenance run. Job C is an engineering/scientific run with high use of processor and memory and very little input or output. If you will follow any one of these jobs on the chart you will see how little the computer system would be used if just that job alone were being processed. In the multiprocessing B 2500 and B 3500 Systems, most of the equipment is used most of the time.

### MAJOR PRIORITY PROGRAMS

Many computer users have several long jobs which must be run at regular intervals on a high priority basis. On conventional equipment these jobs tie up the entire system over a long period of time. Multiprocessing gives the user access to his system during these long runs and allows him to process a number of other programs without disrupting the major run.

### DATA COMMUNICATIONS

The multiprocessing capabilities of the B 2500 and B 3500 lend great power to these systems for profitable data communications applications. Rather than remaining idle while waiting for a message or a request from the data communications network, these systems process the installation's normal work. When a request is received, it is processed simultaneously with the current workload. Priorities are maintained. By eliminating wasted time between requests and spreading system cost among communications and normal processing, the B 2500 and B 3500 help break the economic barriers to many data communications and time sharing applications.

### EXPANSION WITHOUT REPROGRAMMING

When the B 2500 or B 3500 configuration is expanded to include more memory or additional peripheral devices, a control card is entered to advise the MCP of the change. This simple action ordinarily represents the total reprogramming required. The MCP automatically reorganizes the in-process program mix to take full advantage of the system's new capabilities. Changing from a B 2500 to a B 3500 is just as simple and usually requires no reprogramming whatsoever.

### GRACEFUL DEGRADATION

If certain types of equipment must be deactivated or removed from the system for maintenance, the MCP may be advised with a control card. The MCP will then dynamically reorganize processing and allow the system to continue to operate efficiently, although possibly with fewer programs in the mix.

### PROGRAMING SIMPLICITY

Many of the tedious, repetitious operations inherent in programming have been absorbed by the MCP. B 2500 and B 3500 programmers avoid nearly all detail work involved in memory allocation, loading routines, file opening and closing, record blocking and unblocking, I/O procedures, program overlays, library calls and other computer housekeeping. They can concentrate their talents on solving their problems rather than those of the machine.

### EASE OF OPERATION

The self-scheduling B 2500 or B 3500 system takes over many operating tasks, greatly easing the responsibilities of both the operator and the computer department manager. At set-up time, the operator may mount tape files on any convenient, unused, tape unit. The MCP will record the file identification and will assign it to the program at initiation time. The MCP also automatically assigns scratch tapes. The combination of MCP control and extensive disk file storage capabilities makes it possible to store most programs and many files in random access storage. This greatly reduces manual tape reel and card deck filing and retrieval operations.

Nearly all bookkeeping is done by the system, as is most scheduling. With the scheduling problems eliminated and with a detailed accounting of system use and utilization at his fingertips, the DP Manager can work with greater efficiency and ease on new applications and on further improvements of current operations.

# COBOL

Burroughs B 2500/B 3500 COBOL offers all the advantages inherent in U.S.A.S.I. (formerly A.S.A.) COBOL plus the unique benefits of B 2500 and B 3500 hardware design and MCP guided operation. COBOL consists of an English-like, business oriented language and a compiler program which translates this language into computer code. It has been implemented in installations across the country for nearly a decade and is the most widely used business oriented compiler language in existence.

COBOL's popularity has been based on its distinct advantages to the business data processing installation. Its easy-to-read language allows management and supervisory personnel to see just what steps are taken in a program and to understand the process. Debugging and program modification are simplified and documentation is nearly automatic. Since

## HARDWARE/SOFTWARE INTEGRATION

Burroughs B 2500 and B 3500 hardware is built to implement COBOL.

Special micro-operators are included to initiate whole COBOL statements and eliminate much object coding. Consider the statement "MOVE FIELD-A TO FIELD-B". The B 2500 and B 3500 accomplish this move with one object code instruction even when one field is "packed" and the other is not and regardless of the length of the data fields. In addition to the many micro-operators built into the hardware to reduce coding and increase speed, the systems include special registers to facilitate COBOL usage.

Burroughs B 2500/B 3500 COBOL is also

## BURROUGHS B 2500/B 3500 COBOL SAVES TIME AND COST

### REDUCES PROGRAMING AND DEBUGGING TIME

Burroughs B 2500/B 3500 COBOL programs are less intricate and usually shorter than

a COBOL statement may represent an entire string of machine instructions, programs written in COBOL tend to be shorter and simpler; they require less writing time; and they are easier to test. In addition, COBOL programs written for one computer system may, with minor modifications, be run on another system and similarly, programmers are easily retrained to write COBOL programs for different computer systems. All of these advantages are inherent in COBOL usage.

B 2500/B 3500 COBOL meets or exceeds all the standards proposed by the United States of America Standards Institute X3.4.4 task group in May, 1965. This recently standardized version of COBOL has been carefully integrated with B 2500/B 3500 hardware and MCP design to insure optimum operation on these systems.

integrated with the MCP. The MCP enhances COBOL programing, compilation, and object program execution. Since the MCP automatically carries out most housekeeping operations associated with I/O scheduling, file handling, and memory allocation, programing requirements for these tasks are greatly reduced. The MCP also takes over most of the intricacies of overlaying program segments. COBOL compilations may be multiprocessed with other COBOL compilations, FORTRAN compilations, or any object or assembly program suited to the B 2500/B 3500. The resulting object programs may also be multiprocessed.

programs written in lower-level languages. Debugging aids, such as TRACE, MONI-

TOR, and DUMP commands, have been incorporated so that the program can create the finest logic trails available. Additional aids to programing include:

- "COPY" and "INCLUDE" commands that allow immediate access to files and procedure paragraphs included in previous COBOL programs and stored in the program library. Thus the programmer may in one statement retrieve a string of previously written statements and greatly reduce his programing effort.
- Conditional relations and arithmetic expressions may be written algebraically or in COBOL language. Use of algebraic notation often reduces statement length and programing time.
- A full complement of figurative constants helps eliminate program logic errors and extensive coding requirements.

## SIMPLIFIES CHANGE

COBOL's easy-to-read language and standardized format make it almost self-documenting. Modifications to meet new application requirements may be made easily by any COBOL programmer. A new programmer can, with little or no trouble, understand and complete or modify another programmer's work, thus softening the effects of personnel turnover. Equipment changes are also much easier to handle when programing has been done in COBOL. With minor changes, a COBOL program may be re-compiled and run on a new computer system. The change from a B 2500 to a B 3500 system requires no reprograming whatsoever.

## REDUCES TRAINING TIME

COBOL's English-like language lends itself to memorization and understanding. To further simplify training, Burroughs Corpora-

tion has prepared a programed instruction course for B 2500/B 3500 COBOL. Using this programed text, your programmers can master B 2500/B 3500 COBOL basics in approximately 15½ hours. No instructor or special classroom is needed. This technique has proven to be significantly more effective, more rapid, and less costly than traditional teaching methods.

## PRODUCES EFFICIENT OBJECT PROGRAMS

As part of the standardization process, the new COBOL has been revised to eliminate those elements which, in the past, tended to produce inefficient object code. As a result, the new COBOL produces object programs which are approximately equal in efficiency to above-average hand-coded programs. Hardware and control program innovations in the B 2500 and B 3500 add even greater capabilities to this new COBOL.

- Subordinate programs may be called from the program library and multiprocessed with the calling program.
- Data work areas may be renamed and used by successive portions of the program. Total required memory area is reduced by allowing the same area to be used by several procedures within the program.
- Printer output may be detoured to a tape unit when the printer is busy and then routed back to the printer when it is free. This eliminates processing delays and increases system efficiency.
- Tape files may be opened and read in either forward or reverse mode. This ability often eliminates rewind delays.
- Numeric literals may be up to 160 unsigned or 99 signed digits, thus giving B 2500/B 3500 COBOL unsurpassed power and simplicity of digital manipulation.

# FORTRAN

Fortran is offered as the major computational programming system for the B 2500 and B 3500. It is the most widely used compiler system in existence. Fortran is similar in concept to COBOL, but is tailored to scientific and engineering applications. Burroughs B 2500/B 3500 Fortran is adopted from A.S.A. (now U.S.A.S.I.) standards set by the X3.9 task group and accepted March 7, 1966.

Fortran offers two main advantages to the scientific/engineering computer installation: improved communications and greater programming speed. Its language is expressed much like mathematical formulae and is easily learned by most scientists and engineers. Once learned, Fortran allows these people to communicate directly with the computer system for many of their processing requirements and to work closely with the programming staff on more complicated applications. The second major Fortran advantage, speed, results from its simplified condensed nature. Each Fortran statement represents a string of machine instructions. The programmer does not have to concern himself with these machine instructions but instead uses the Fortran language as a kind of shorthand. The result is shorter, more straight-forward

programming.

Statements written in the Fortran language are translated into machine language by the Fortran compiler. On the B 2500 and B 3500 systems, this translation, or compilation, is done under the guidance of the MCP and may be multiprocessed. The resulting program is usually about equal in processing efficiency to one devised in machine language by a competent programmer. The time and cost involved in producing a Fortran program, however, is far less than that of machine coding.

The standardization inherent in Fortran programming has several major advantages. A program does not have to be returned to its author to be revised. Any programmer with Fortran training can understand another programmer's Fortran program and expand it, revise it, or adapt it to another computer system. This flexibility can drastically cut costs during normal operating conditions and offers even greater savings when a new system is installed and software conversions must be made. Little or no reprogramming is required when changing B 2500 or B 3500 configurations or moving to another computer system.

## SOME BASIC FEATURES OF B 2500/B 3500 FORTRAN ARE DEFINED BELOW:

- Integer length: 1 to 99 decimal digits.
- Maximum integer magnitude:  $10^{99}-1$ .
- Floating point precision: 1 to 96 decimal digits.
- Maximum floating point magnitude:  $10^{195}$ .
- Precision of double precision arithmetic: up to 96 decimal digits.
- Maximum number of array dimensions: 3.
- Arithmetic mode: Mixed floating point and integer.
- Eight different format specifications.

Type declarations:

INTEGER

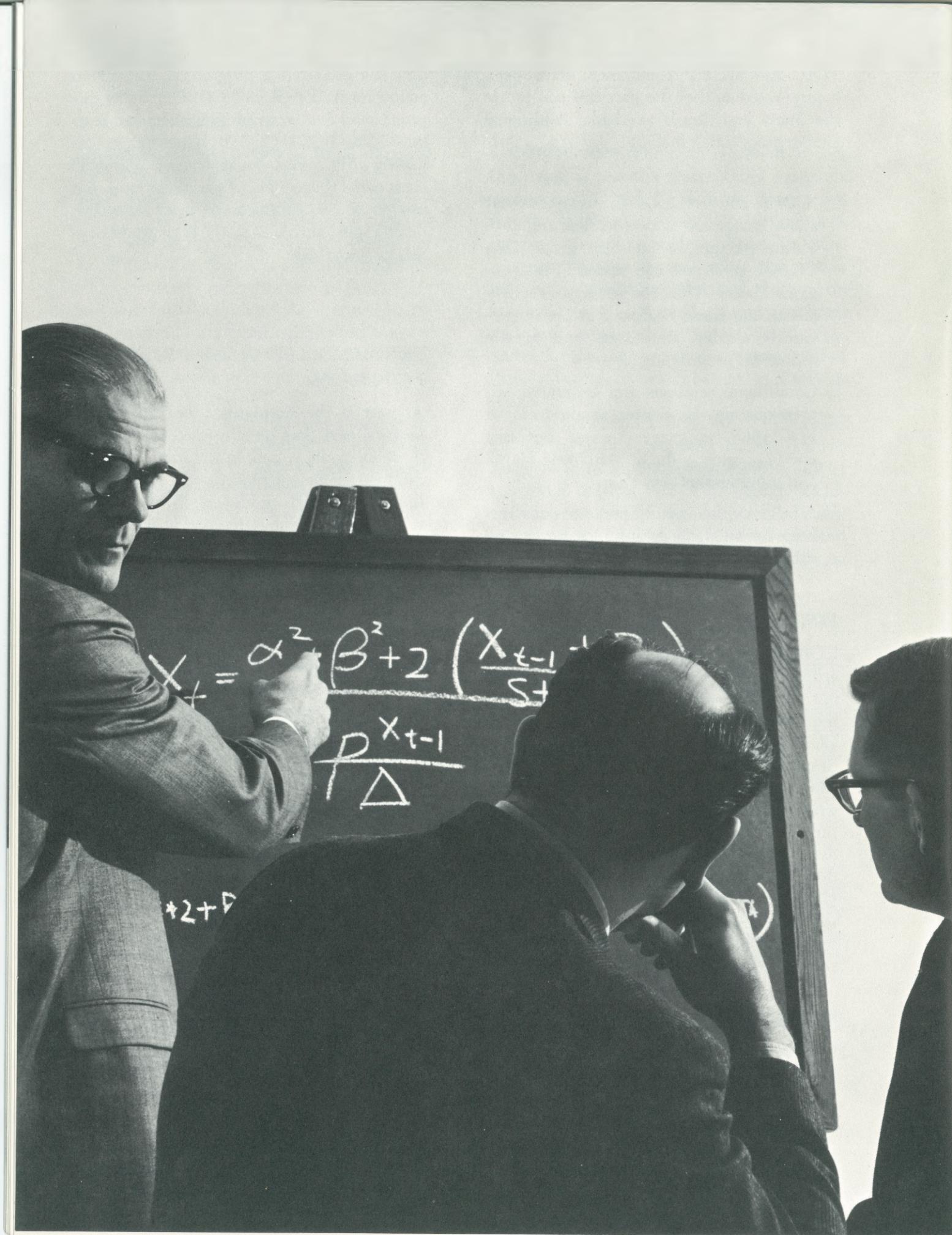
REAL

HOLLERITH

LOGICAL

DOUBLE PRECISION

EXTERNAL



## ASSEMBLER PROGRAMING SYSTEMS

Assembler Programing Systems offer B2500 and B3500 users the flexibility inherent in symbolic languages plus the advantages of many compiler-like macro instructions to further ease the programing task. The languages include facilities for symbolic addressing, diagnostics, and operation with the control programs. The Basic and Advanced Assemblers work with the Basic Control Program and Master Control Program respectively.

While coding in the symbolic Assembler Language, the programmer has complete flexibility of instruction modification, indexing, character or bit manipulation, and program segmentation.

Programs coded in the Assembler Language are translated into machine language by the Assembler Program. The time required for this assembly process is minimal, and, on MCP controlled systems, assembly operations may be multiprocessed. The resulting object programs operate in concert with the control programs to take full advantage of the hardware system capabilities and operate at optimum speed.

### ASSEMBLER LANGUAGE

Assembler Languages are designed to make programing easier and more profitable to the installation. Language features include:

- Symbolic addressing, and increments and decrements of these addresses.
- Up to six-character symbolic labels.
- Self addressing, literal insertions, and machine absolute coding techniques.
- Macro instructions to minimize standard routine writing.
- User-inserted macro instructions.
- Pseudo operating codes to control the assembly process.
- Use of alphanumeric literals within arithmetic, logical, and data movement instructions.
- Symbolic reference to three index registers per program.
- Data declaration to declare contiguous tables, (arrays) of data, constants, punctuation for editing, and constants to be used for indexed addressing.

### ASSEMBLER PROGRAMS

The Assembler Programs are provided to translate the programs written in symbolic Assembler Language into highly efficient machine language object programs. Ease of use is enhanced by the Assembler Program's ability to handle:

- Variable length instructions.
- Indirect addressing.
- Indirect field length.
- Source program input media of punched

- Data definitions for each address (alphabetic, packed numeric, signed or unsigned).
- Segmentation facilities.
- Diagnostic facilities indicating source language errors.

In addition, the Advanced Assembler includes:

- An unlimited number of symbolic labels.
- An unlimited number of program points.
- File declarations to identify hardware labels, recording mode, buffer areas, retention period, record blocks and buffer access techniques.
- Macro instructions for opening and closing files and associated file checking and for testing and initiation of I/O operations.
- Macro instructions to allow the programmer to control the vertical formatting of the printed assembler listing and to list selected portions of the program.

card, punched tape, or magnetic tape, or, with the Advanced Assembler, disk file.

- Corrections introduced through paper tape or punched card.
- Detection of illegal codes, illegal addressing, and incorrect designation of data fields while assembly progresses.
- Machine language output to the printer, magnetic tape, punched cards, or punched tape, or, with the Advanced Assembler, disk file.

## REPORT PROGRAM GENERATORS

The Report Program Generators provided with the B2500 and B3500 systems allow the user to produce many programs of the simpler variety in minimum time and at optimum efficiency. The Report Program Generators are generator programs which compile relatively complex symbolic programs from a brief, simplified, problem-oriented language. Program generation is fast and the resulting programs will normally run at the rated speeds of the designated peripheral equipment. Report Program Generators are provided for use with either the Basic or Master Control Program and both the generative programs and the resulting report programs may be multiprocessed on an MCP controlled system.

### HERE ARE A FEW REPORT PROGRAM GENERATOR CAPABILITIES:

- Object programs accept input from magnetic tape, punched tape, punched cards, and disk file storage (advanced version only). A maximum of three types of input, in any combination, is permitted.
- Object programs produce output for up to four of the following: magnetic tape, punched tape, punched card, printer, or disk (advanced version only). Only one printer is permitted but other types of output may be used in any combination.
- Input/output formats are unrestricted.
- Up to nine levels of totals are permissible.
- Up to thirty accumulators are available for each total level.
- Automatic "counter rolling" from each class to the next is provided.
- Printer output can be edited to conform to 100 unique, user designated "pictures".
- Maximum length of input/output records and blocks is 9999 digits for numeric data and 9999 characters for alphanumeric files.
- One hundred constants may be designated by user.
- The user may specify complete forms control both before and after each printed line. Up to 27 unique user-designated paper motion descriptors may be provided for printing detail, headings, and total lines.
- Up to 100 halts may be designated.
- Tabulated or detail listed reports may be produced.
- Class control is allowed on an unlimited number of fields.
- An unlimited number of fields within an input record can be identified.
- Ability to subtract, based on characters other than the X punch, is provided.
- Crossfooting on a detail record basis is included.
- Selective multiple-line printing capabilities are provided.
- Dual line headings with page counting are available.
- Sequence checking can be done on the basis of ascending or descending sequence, equality, non-equality, or any combination thereof.
- Input data, constants, working storage, and accumulators can be inserted into headings and other print lines, subject to any set of user-designated conditions.
- An operator is included which will convert a sterling field to pence. Another operator will invert an all-pence field to pounds-shilling-pence.
- Selection of normal and auxiliary stackers can be effected under program control at object time.
- Generated symbolic coding may, at the user's option, be printed at generation time.

# SORT PROGRAM GENERATORS

Sorting procedures, which take 30 to 50 percent of total run time in many installations, are simplified and made more efficient by the generative programs provided with the B 2500 and B 3500.

These Sort Program Generators are designed to reduce programing time and to quickly generate programs tailored to the user's requirements.

The Basic Sort Program Generator is magnetic tape oriented and operates in conjunction with the Basic Control Program (BCP). The Advanced Sort Program Generator is available in both magnetic tape and disk file versions and operates with the Master Control Program (MCP).

## ADVANTAGES OF SORT PROGRAM GENERATION

### PROGRAMING SIMPLICITY

Input to these generative programs consists of a few specification cards defining the sort requirements and the equipment complement to be used. From this easily prepared input, the generative programs produce fast running, highly efficient sort programs tailored to the specific application.

### SPECIAL ROUTINES

Special routines may easily be inserted into generated programs to meet unique requirements. An optional symbolic output provides a well documented listing of the generated program and facilitates additional coding.

### EFFICIENT RESPONSE

Generated programs are in symbolic language and may be punched into cards or paper tape, or written on magnetic tape or disk file. The Advanced Sort Program Generator allows the user to have the generated program assembled and assigned to the program library on magnetic tape or in disk file. Then, one call card or one entry through the console keyboard will call the program into memory where it may immediately be used in processing.

## TYPES OF PROGRAMS GENERATED

### FULL RECORD SORT

Programs produced by the Sort Program Generators are divided into the stringing phase, intermediate string merging, and final merge. Incoming data is sorted by a vector replacement technique and the sequenced strings are distributed on the work tapes through a Fibonacci distribution.

A series of merge passes on the sequenced strings produced during the first phase creates longer strings until the number of strings remaining is less than or equal to the order of the merge. During the merge phase the blocking factors may be changed in order to use all available memory.

The last phase performs the final merge pass and distributes the sequenced file to magnetic tape or disk file. During the last phase, output records are reblocked if necessary to comply with the sort specifications.

## TAG SORT

Tag sorting, available with the Advanced Sort Program Generator, and disk file system configurations, greatly reduces run time by moving and manipulating only a small portion of each record. The entire record is placed in the disk file. The identification portion of the record essential to sorting is combined with the record's disk file address, and is read into core memory to form a "miniature record". All sort operations are performed on this "miniature file". When it is properly sequenced, the miniature file is used to access the complete records in the disk file in the desired order. Use of the miniature file reduces the amount of data which much be read, moved, and otherwise manipulated. It thus diminishes core memory requirements, disk file accesses and run time.

## BASIC FEATURES

### THE SORT PROGRAM GENERATORS:

- Permit a sort key length of up to 100 characters.
- Allow sort key to be split among ten different areas of the record.
- Permit ascending or descending sequence of records to be specified.
- Allow merge-only operations.
- Produce generated programs in symbolic language.

### THEY PRODUCE PROGRAMS WHICH:

- Accept either fixed or variable, blocked or unblocked records as input.
- Block records in merge phase for greater efficiency.
- Use read reverse logic to eliminate rewinding working tapes.

- Have automatic restart procedures.
- Handle input, blocked or unblocked, of any size within main memory limitations.
- Provide record counting verification.
- Permit multiple-reel input and output.
- Provide file buffering for read/write/compute capabilities.
- Perform label checking for standard Burroughs labels.

### ADVANCED SORT PROGRAM GENERATOR PROGRAMS ALSO:

- May be multiprocessed during generation and sorting.
- Provide tag sort capabilities.
- Have automatic overflow to tape from disk file.
- Recognize sign control when sorting arithmetic fields.

# MEDIA CONVERSION PROGRAM GENERATORS

The Burroughs B2500 and B3500 Electronic Data Processing System user has the advantage of generative data conversion software. Generation eliminates the cost of individually programing conversion routines—yet provides better flexibility and greater efficiency than is possible with standard routines. The Media Conversion Program Generators, with a single specification card input, provide complete media conversion routines tailored to individual requirements. Both the generative programs and the generated media conversion routines take full advantage of the computer system's speed and capabilities.

## PROGRAM GENERATION

### INPUT

A job specification card is the only input required by the Media Conversion Program Generators. The peripherals to be used for input and output, whether records are to be blocked or unblocked, and all other specification details are coded on this one card and read into the system to initiate the media conversion program generation.

### PROCESSING

The Media Conversion Program Generator is stored as part of the software system library and is called into action by the specification card. Processing takes place under control of the Basic or Master Control Program.

### OUTPUT

The Media Conversion Program Generators will:

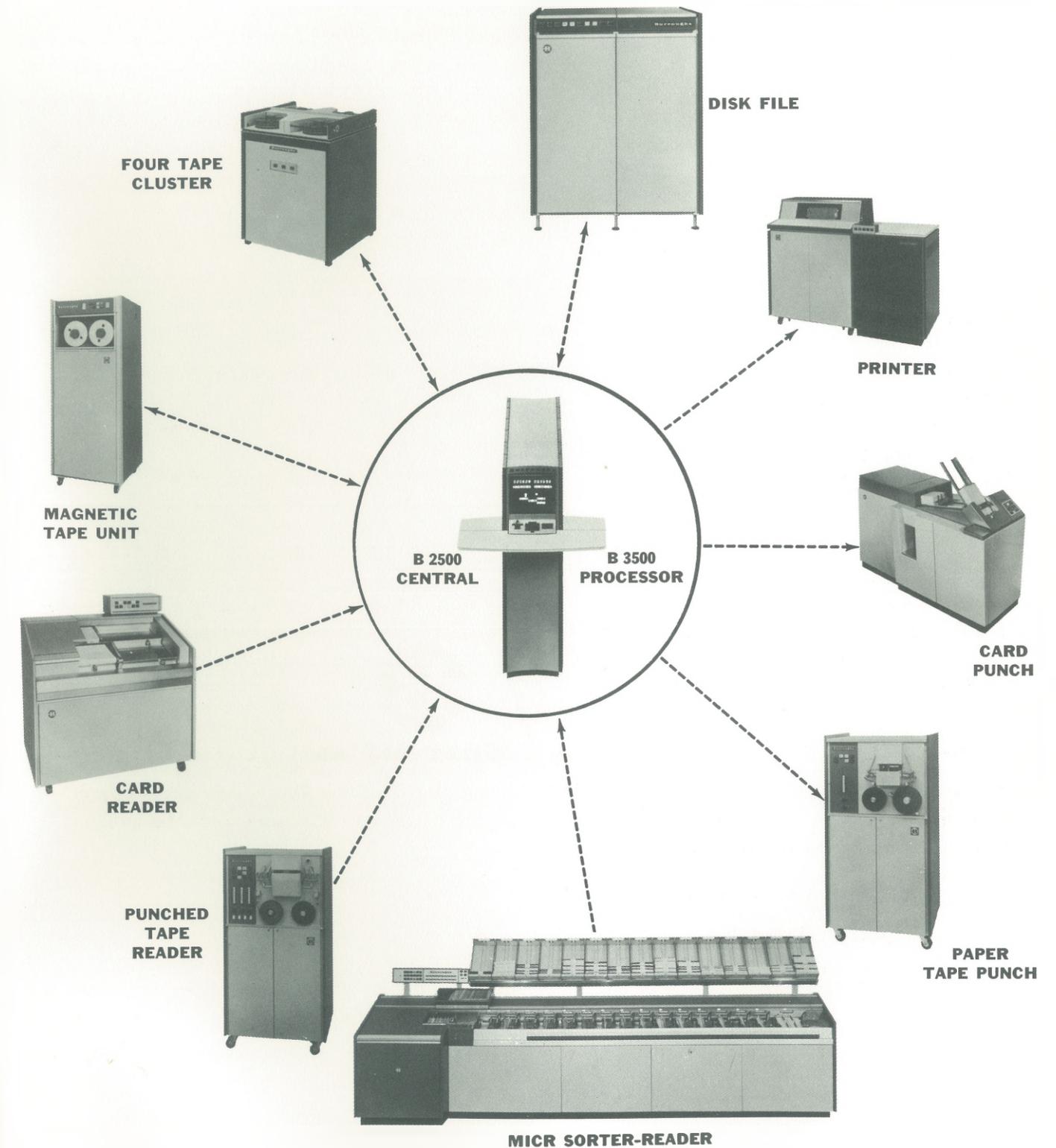
- Generate the media conversion program and allow it to start processing data immediately on a generate and go basis.
- Generate the program and store it in the software library for future use.
- Generate the program and list it in symbolic language on the printer. The symbolic listing makes it a simple operation for the programmer to add processing routines to the program and tailor it still more, if necessary.

## ADVANTAGES OF THE MEDIA CONVERSION PROGRAM GENERATORS

The Media Conversion Program Generators offer speed and flexibility at low cost in operations commonly held rigid by standard routines or requiring expensive programing. The Media Conversion Program Generators provide:

- Programs tailored to the application's requirements.
- Easy modification procedures.
- Shorter programs.
- Generate and go capability.

## MEDIA CONVERSIONS POSSIBLE WITH GENERATED PROGRAMS





*Wherever There's  
Business There's*



**Burroughs**