DISTRIBUTION LIST

B1800/B1700 SOFTWARE PRODUCT SPECIFICATIONS

DETROIT

T. Freeman - Prod. Mgmt.
F. D'Angelo - PROD. MGMT.
J. Dormady - International
W. Varns - BMG
L. Thomas - BMG
J. M. Ross - Int'l Group P
B. Dent - CSG
D. Dahm - Corp. Eng.
W. B. Hunt - SSG
P. E. Pleming - Int'l FE
D. Hill - TC, BM, & SS

U.S. AND EUROPE

S. Oksalo - (Plymouth)
R. H. Mack - TSG (Plymouth)
J. H. Pedersen (Plymouth)
J. Berta (Downingtown)
W. Minarcik (Paoli)
G. Smolnik (Paoli)
R. Boruta - F&SSG (Paoli)
J. Murtaugh (Tredyffrin)
A. Kosla (McLean)
A. LaCivita - F&SSG (McLean)
R. Ledford - F&SSG (McLean)
R. Sutton - F&SSG (McLean)
Mgr, WADC (Irvine)
W. H. Keil (Pasadena)
H. M. Townsend (Pasadena)
N. Cass - Pat. Atty. (Pasadena)
D. C. Swanson (Mission Viejo)
H. O. Wickman (Mission Viejo)
J. J. Dowling (Westlake)
H. N. Riley (El Monte)
W. E. Feeser (Austin)
J. C. Allan (Glenrothes)
W. McKee (Cumbernauld)
I. J. Carradine (Cumbernauld)
Mgr, NPSGrp (Ruislip)
P. R. Evans (Middlesex)
B. Hammersley (Croydon)
J. Gerain (Pantin)
J. Cazanove (Villers)
J. C. Wery (Liege)
R. Bouvier (Liege)
S. Samman (Liege)
C. J. Tooth - SSG (London)

SANTA BARBARA PLANT

R. S. Bunker
J. Hale
R. Shobe
K. Meyers
A. van der Linden
T. Cardona
R. Bauerle
J. Henige
E. Yardi
J. Darga
B. Ross-Smith
E. Munsch - 2
G. Hammond - 2
J. Morrison - 6

Distribution list current as of 05/14/79

Burroughs Corporation
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

PLANT IDENT	0 1 4

DOCUMENT TRANSMITTAL

PAGE ____ OF ____

CONTROL DOC.	REV
None	

SYSTEM/PRODUCT B1800/B1700 B500IEP DISTRIBUTION TYPE ____

UNIT NAME ____ PROJ. NO. ____

COORD. ____ CHKR. ____ RECORDS ISSUE DATE 6-12-71

DOC PFX OR DWG SIZE	DOCUMENT NUMBER	REV OR ISSUE	DOCUMENT DESCRIPTION	ENG. RECORDS ONLY				REMARKS
				QTY REV PAGE	COPIES EACH PAGE	BOND	VELLUM	
P.S.	2222 2731	B	B500IEP	26	65			
DOCUMENT TRANSMITTAL								

Burroughs Corporation

COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

B1800/B1700 B500 INTERPRETER ENVIRONMENT PROGRAM

PRODUCT SPECIFICATION

REV LTR	REVISION ISSUE DATE	APPROVED BY	REVISIONS
A	3/2/78		Original issue, Software Release Level Mark 7.0.
B	4/9/79		MARK 8.0 RELEASE ADDED TAPE.RETRIES SECTION TO OPTIONAL DECLARATIONS DIAGRAM. ADDED 'TAPE.RETRIES OPTION WILL' PARAGRAPH. ADDED OPTIONAL SPO INPUT PARAGRAPH.

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

COMPANY CONFIDENTIAL
B500 IEP
P. S. 2212 5113 (B)

TABLE OF CONTENTS

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

INTRODUCTION

GENERAL

The B500 Interpreter Environment Program (IEP) is used to establish an environment for the B500 Interpreter since the B500 Interpreter requires an environment file that describes the memory size, peripheral configuration and special options of the target B500 environment. Establishing an environment file parallels the compilation of a programming language except that there is no source code read nor object code produced. The B500/IEP accepts free format specifications and uses default characteristics wherever possible. The environment file, when written to B1800/B1700 disk, may be used for B500 interpretation until the configuration of the B500 system changes, and then the environment file must be recompiled, or modified.

After the environment is established, the environment file is executed to start B500 interpretation. The execution phase is discussed in another product specification, B500 Interpreter (P.S. 2212 5348). This product specification describes the compilation phase and the requirements of the interface to the execution phase. The discussion is supplemented with examples and an appendix. The alphabetic index references the major components of this program.

RELATED PUBLICATIONS

Name	Number
B500 Interpreter	P.S. 2212 5348
B1800/B1700 Software Operational Guide	1068731

SYNTAX DEFINITIONS

B500/IEP instructions are displayed through railroad syntax diagrams, a concise method of explaining syntax that involves defaults, alternatives, and iterations. The basic rule of this method is that any path traced along the forward directions of the arrows will produce a syntactically valid command.

B500/IEP instructions are constructed of letters, digits, special characters and blanks. Alphanumeric characters may be separated into such items are integers, keywords and identifiers.

KEYWORDS

Upper-case letters in syntax diagrams indicate keywords which must appear literally in the command.

BLANKS

Blanks in B500/IEP commands serve to separate syntactic items and may appear freely except within certain text fields, where they become significant characters. Whenever one alphanumeric item (keyword, identifier, integer, etc.) follows another, they must be separated by at least one blank.

SYNTACTIC VARIABLES

Lower-case letters, words and phrases that occur within broken brackets (<>) are syntactic variables, and these variables represent information to be supplied by the user. A particular variable may require a single character (e.g., <letter>), a simple construct such as an integer or text string (e.g., EOJ = 9<mn>), or a relatively complicated construct (e.g., <disk.name>). The basic variables used in B500 IEP syntax are defined as:

<number> A positive integer.

<digit> A positive integer between zero and F.

<character> Abbreviated as <char>.

<name> A textual string consisting of: one to ten characters, no imbedded blanks, beginning with an alphabetic character.

<pack-id> A <name>, i.e., from one to ten characters with no imbedded blanks.

<multifile-id> A <name>, i.e., from one to ten characters with no imbedded blanks. Abbreviated as <mfid>.

<file-id> A <name>, i.e., from one to ten characters with no imbedded blanks.

<disk.name> From one to three <name>s, consisting of a <name>, a <mfid>/<name>, a <pack-id>/<mfid>/ or a <pack-id>/<mfid>/<file-id>. The term can be defined through railroad syntax as:

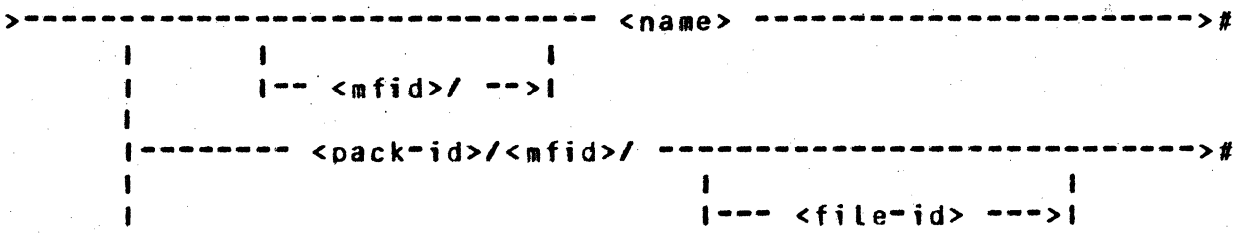


Figure 2.1 Sample Syntactic Variable

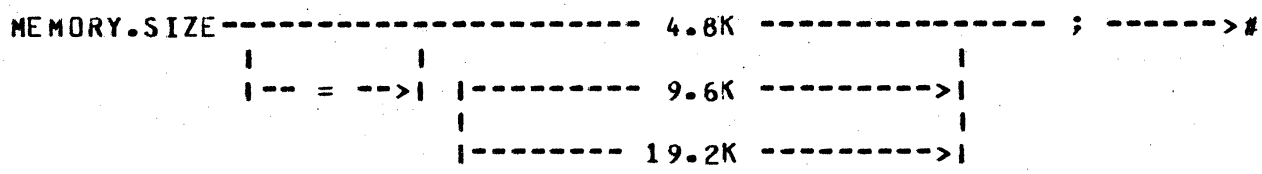
ENVIRONMENT: DEFINED

The environment file is created by specifying the memory size and peripherals that the B500 Interpreter will use to process data in the target environment. As the syntax will show, each specification contains required and optional entries. When a device specification is omitted, it is assumed that the corresponding device will not be part of the emulated environment. If a device specification does not define all of the possible characteristics of the device, the IEP will use default characteristics appropriate for that device. Defaults are specified for each device.

MEMORY SIZE

This statement specifies the amount of memory to be included in the emulated environment. This statement is required in every environment specification.

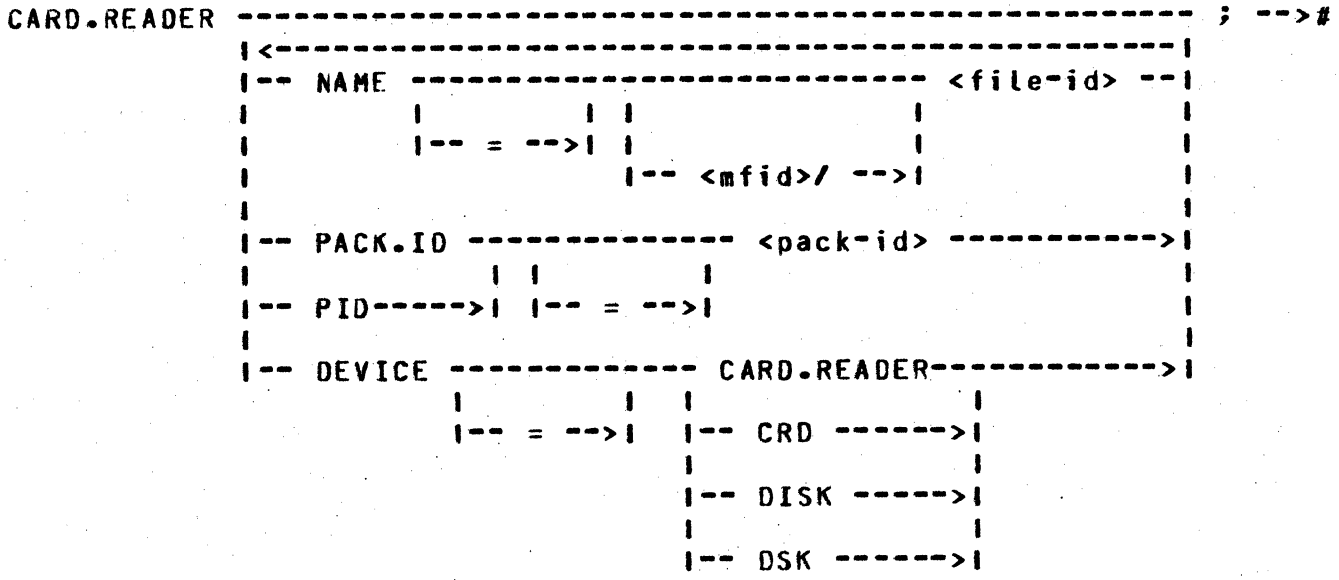
Syntax:



CARD READER

This statement specifies that a card reader with the described attributes is to be included in the environment.

Syntax:



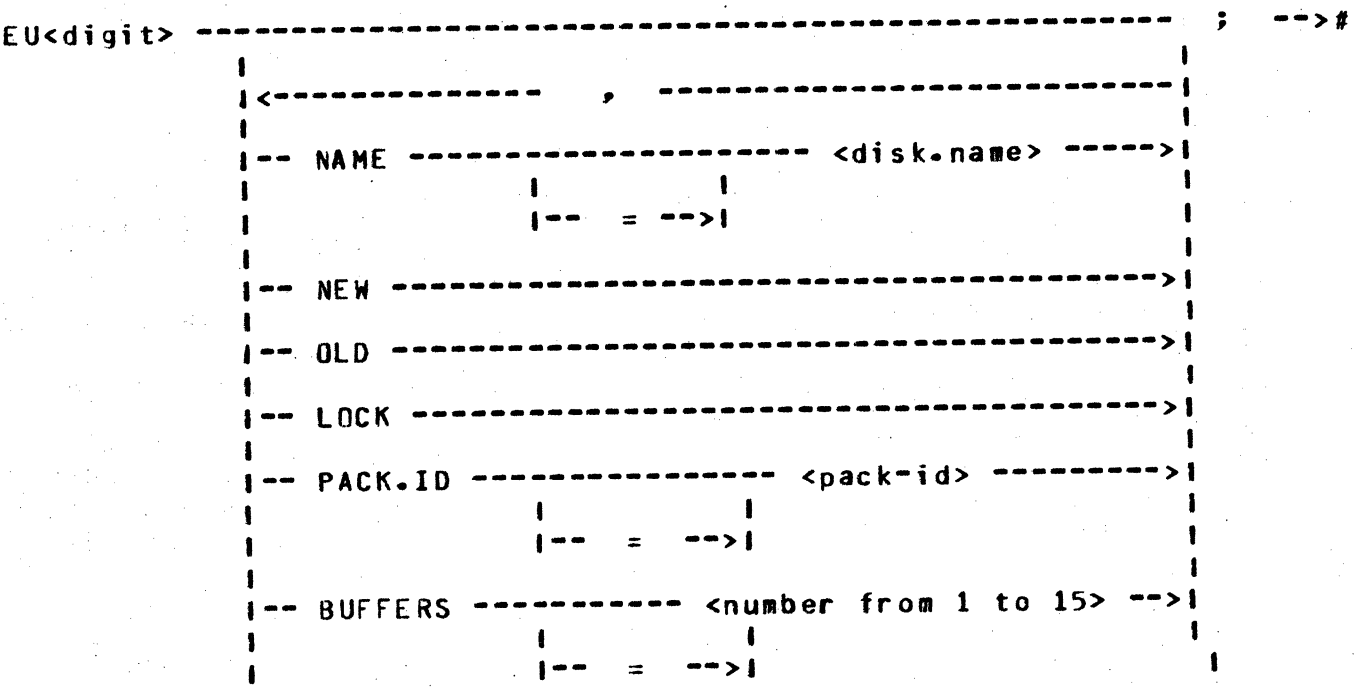
Semantics:

The default name is B500.CARDS and may be changed by the NAME attribute.

EU

This statement specifies that the disk named in the specification with the described attributes is to be included in the environment.

Syntax:



Semantics:

The default name is B500/EU<digit>, and no two EUs are allowed to have the same <disk.name>. NEW specifies that the EU does not exist on the B1800/B1700 disk or, if one exists, it is not to be used. OLD specifies that the EU exists on the B1800/B1700 disk. OLD is default. LOCK specifies that the EU is to be saved in the B1800/B1700 Disk Directory at the end of the interpretation run and any old EU will be replaced. LOCK is reset by default. PACK.ID allows the EU to be placed on a user pack or cartridge. The default is system disk.

BUFFERS OPTION

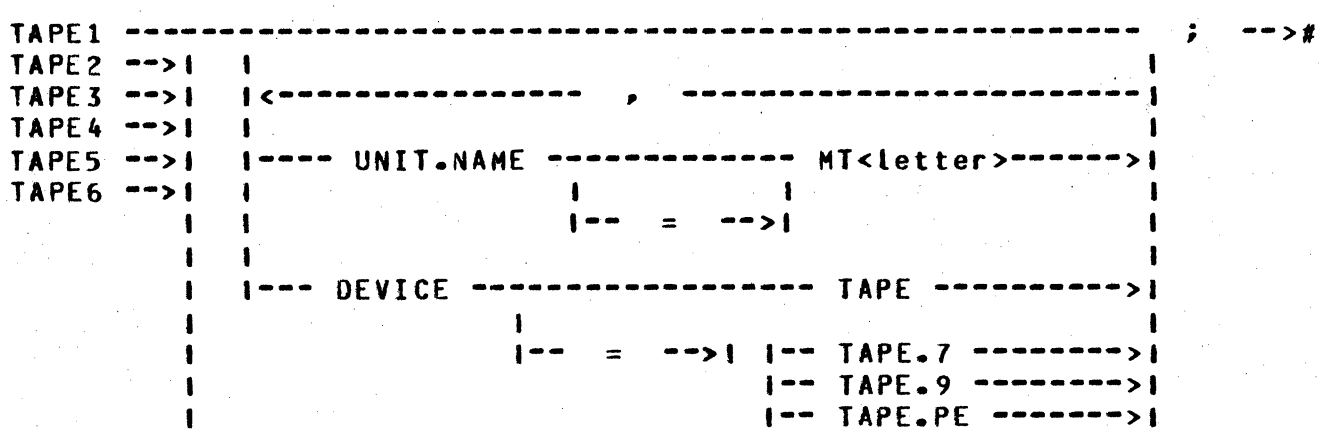
The BUFFERS option allows users to optimize the speed of their disk accesses. Since all types of EU's are RANDOM to the Interpreter, it cannot take advantage of buffer usages for sequential or serial files. As of the Mark VI.1 release, the Interpreter will take advantage of DELAYED.RANDOM, a new disk access method. WHEN DELAYED.RANDOM is used, blocks are only written to disk (an actual I/O) when the buffer is required for storage of a different block. In the original implementation of RANDOM access, an I/O was initiated whenever the Interpreter performed a WRITE communicate.

The number of buffers assigned by the MCP to each virtual disk file is dynamically left up to the user. The default is only one buffer, but the user may specify up to fifteen. It is not necessarily true that a program will run faster if more buffers are allocated; but, depending on the type of disk activity, an additional buffer may make substantial amounts of improvement.

TAPE

This statement specifies that the tape drive named in the specification with the described attributes is to be included in the environment.

Syntax:



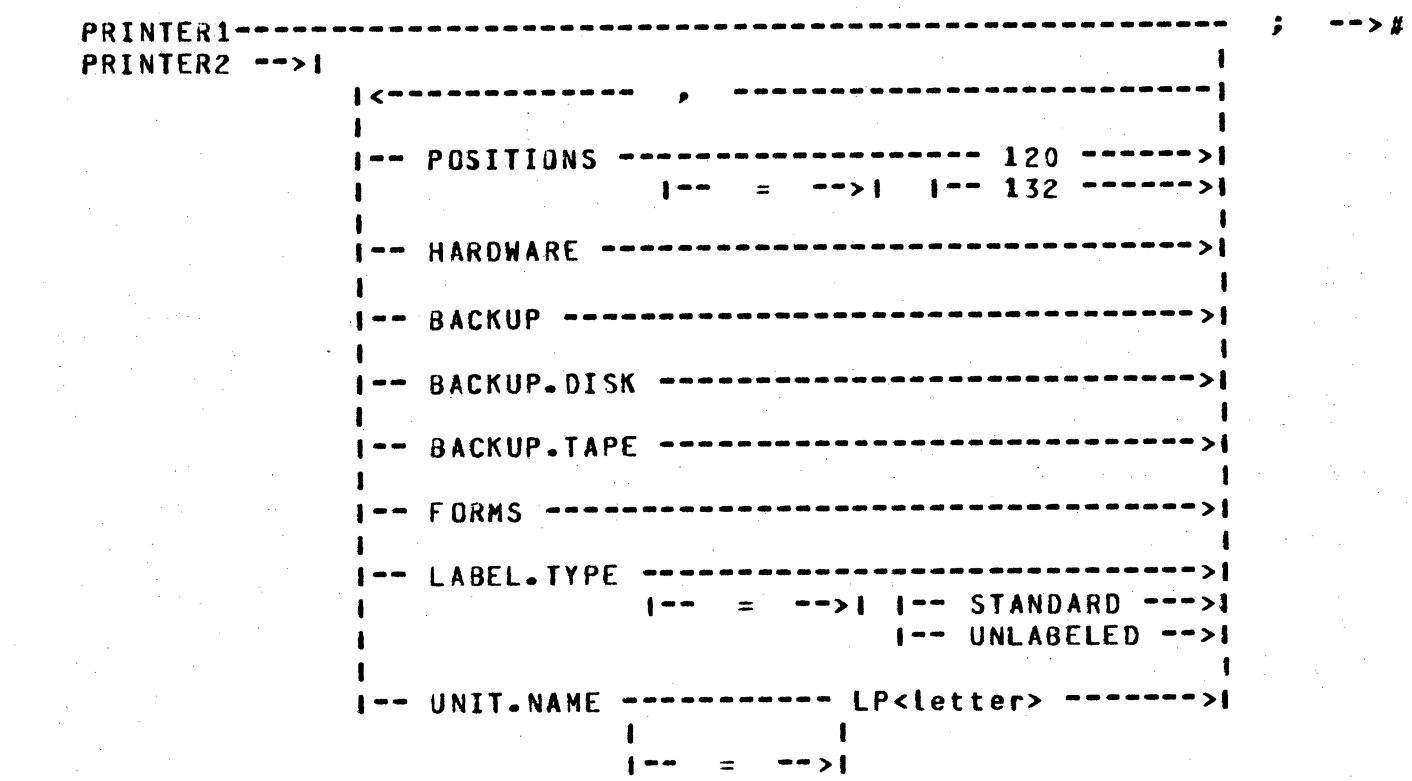
Semantics:

UNIT.NAME specifies that the B1800/B1700 unit named is to be used for the specified B500 tape unit. UNIT.NAME is unassigned by default and the first time the B500 unit is accessed the operator will be requested to assign a unit to the B500 environment. DEVICE specifies the type of hardware to be used for this unit. The DEVICE default is TAPE.7.

PRINTER

This statement specifies that the printer named in the specification with the described attributes is to be included in the environment.

Syntax:



Semantics:

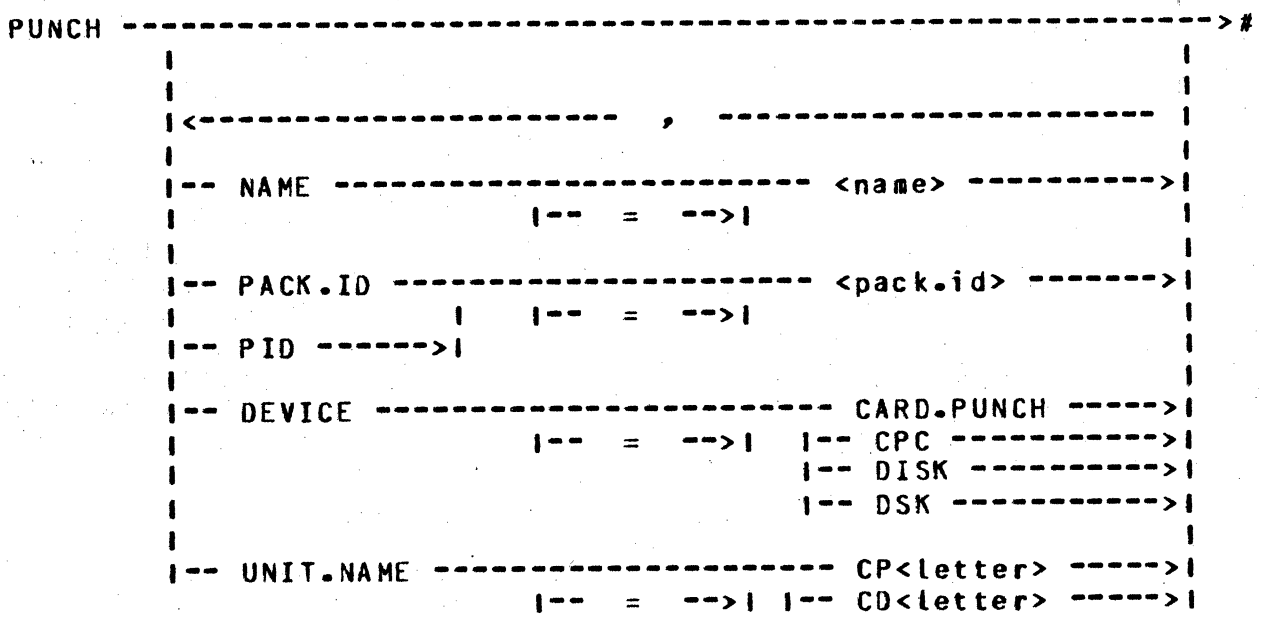
POSITIONS specifies the number of print positions of the printer to be used. If 132 positions are requested and the file is sent to a printer with only 120 positions, the last 12 positions will be lost. 132 is the default. HARDWARE specifies that the file is to go to a physical printer is possible. HARDWARE is true by default. BACKUP will set BACKUP.DISK and BACKUP.TAPE. BACKUP.DISK specifies that the file is to go to BACKUP on disk if a printer is not available. BACKUP.TAPE specifies that the file is to go to backup on tape is a printer is not available.

BACKUP.DISK and BACKUP.TAPE are false by default. FORMS specifies the printer is to be loaded with special forms and will not start printing until the operator loads the forms. FORMS is false by default.

PUNCH

This statement specifies that a punch with the described attributes is to be included in the environment.

Syntax:



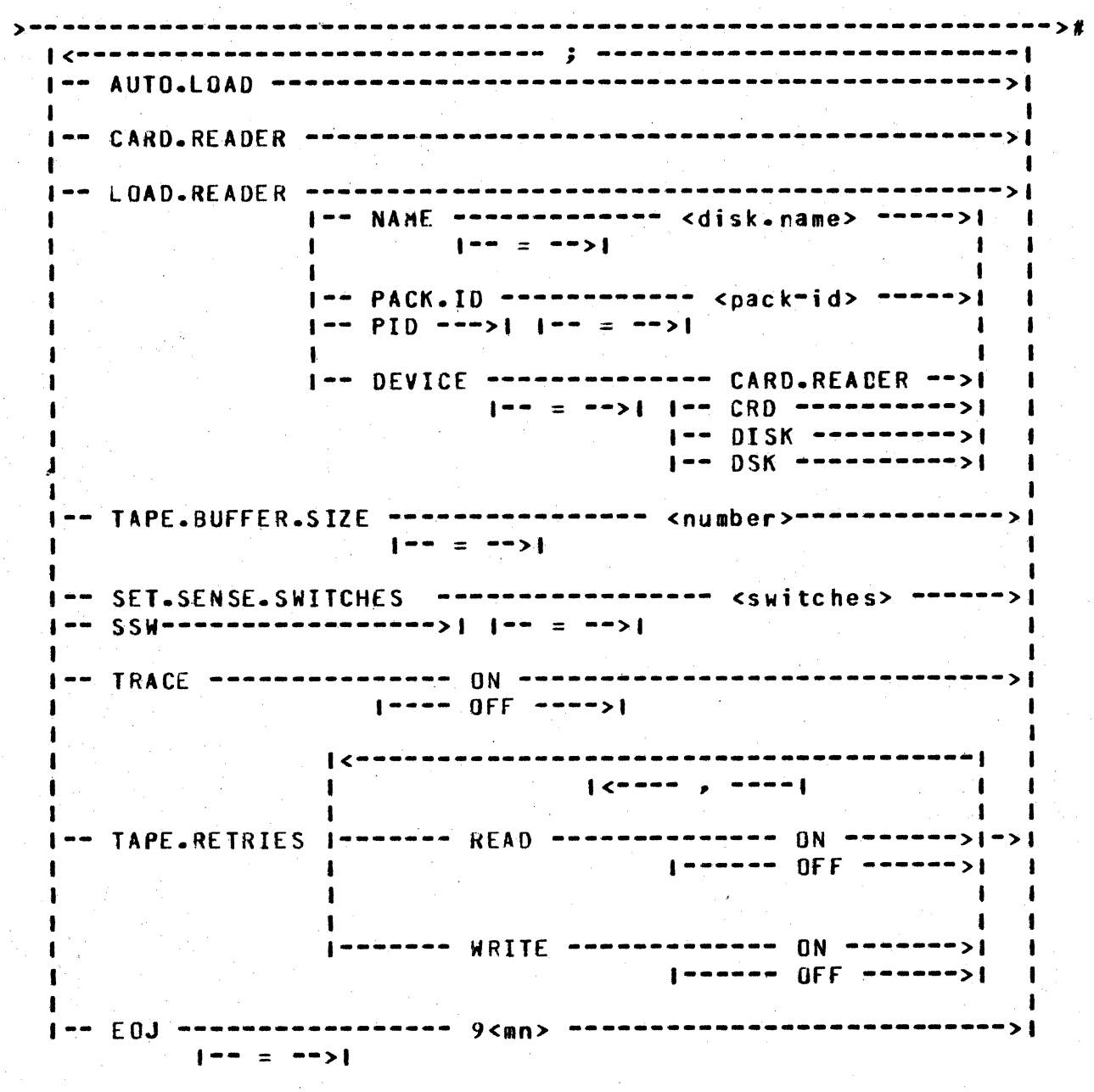
Semantics:

The default NAME is B500.CARDS and may be changed by the NAME attribute. The remaining attributes are the same as for the printer.

If a PACK.ID is specified through "PACK.ID" or "NAME" and DEVICE is not set to DISK, a warning will be issued. NAME may include PACK.ID.

OPTIONAL DECLARATIONS

The optional declarations allowed in a B500 environment are declared as follows:



Notes:

If a PACK.ID is specified through "PACK.ID" or "NAME" and device is not set to DISK, a warning will be issued. "NAME" may now include PACK.ID.

TAPE.BUFFER.SIZE is the physical size of the users tape buffer. It should be slightly larger than the largest tape record that will be read. Note, the size is given in "characters" and automatically "adjusts" to the tape type. The maximum specified number of characters should not exceed the specified MEMORY.SIZE. Note, also that space will not be generated for this buffer unless at least one unit is declared.

SET.SENSE.SWITCHES allows the user to set particular sense switches at BOJ time. The variables <switches> must be sense switch numbers 1 through 6. Any or all of them may be specified and will be set. If they are not specified, they will be reset. The switch numbers must not have blanks or commas between them. For example, "SSW 146;" sets switches 1, 4, and 6.

TRACE causes the interpreter to begin tracing code from BOJ time.

TAPE.RETRIES option will cause the B500 Interpreter to do standard automatic read/write tape retry operations.

```
EX:
     TAPE.RETRIES  READ ON  WRITE OFF;
  or TAPE.RETRIES  READ ON;
```

will cause tape read retries to occur but not tape write retries.

If TAPE.RETRIES is not specified, then responsibility for read/write tape retry operations is left up to the user program.

EOJ option will cause the B500 Interpreter to terminate the B500 virtual machine in a normal fashion. If EOJ has been specified, and the B500 virtual machine executes a halt machine instruction (9<mn>), the <mn> variants of the halt instruction are compared to the <mn> variants of the optional EOJ specification. If the variants are equal, the B1800/B1700 job will terminate.

Note: <mn> can consist of any legal B500 character.

COMPILATION

The B500/IEP creates an environment file using a standard B1800/B1700 compile deck to specify the options desired by the user. Normally, this deck includes the following cards: compile, label (with label equation), data and end. This section lists and explains the various options that can be selected through a B500/IEP compile deck. A sample deck is shown in Figure 5.1.

COMPILE

The compile card is normally the first card in the deck. If it is not included, the operator will have to enter the compile statement from the SPO. The options available are as follows:

OPTION	INSTRUCTION
1. Compile and go	?COMPILE program-name WITH B500/IEP
2. Compile for syntax	?COMPILE program-name WITH B500/IEP SYNTAX
3. Compile to library	?COMPILE program-name WITH B500/IEP LIBRARY
4. Compile/go, library	?COMPILE program-name WITH B500/IEP SAVE

Note: COMPILE may be abbreviated CO; WITH is optional.

LABEL EQUATION CARD

In order to avoid duplicate file names, the label equation card is used, especially when operating in a multi-programming environment. The format of the label equation card is the following:

?FILE <internal-file-name><file attributes>. . .;

If used, the label equation card must preceed the MCP label control card. Refer to Figure 5.1.

The internal file-names and external file-id's used by the B500/IEP program are as follows:

INTERNAL FILE-NAME	EXTERNAL FILE-ID	DESCRIPTION
CARDS	CARDS	Input file from the card reader. The default input is from the card reader.
LINE	LINE	Source output listing to the line printer.
CODE	CODE	Result and codefile will be renamed from the compile card.

EXAMPLE: LABEL EQUATION

To compile an environment program from a disk file with the file-id of "TEST2", use the following deck:

```
?COMPILE B500 WITH B500/IEP TO LIBRARY
?FI CARDS NAME TEST2 DISK DEFAULT;
```

```
    /-----------------------------------------
    ! ?FI CARDS NAME TEST2 DISK DEFAULT;
/-----------------------------------------
! ? CO B500 WITH B500/IEP TO LIBRARY
!
```

Figure 4.1 Sample Compile Deck

MCP LABEL CARD

The MCP label card is formatted as follows:

? DATA CARDS

The card indicates EBCDIC source language input. If absent, the following message will be displayed on the SPO:

**NO FILE<file-name><program-name> = <job-number>

Without specific instructions from the programmer, the operator will not know the proper IL message to give the MCP.

OPTIONAL SPO INPUT

Since the B500 IEP is a program that is run infrequently, and because it is necessary to have a one time special environment occassionally, an optional SPO input instead of card input is available for B500 specifications by setting SW0=1.

EX:

? CO <program name> WITH B500/IEP SW0=1 ⟨SYNTAX / LIBRARY / SAVE⟩

SAMPLE ENVIRONMENT SPECIFICATIONS

If a B500 Interpreter environment called "B500/TEST" is desired, the following characteristics are specified in the Interpreter environment:

1) 9.6K memory

2) 132-character printer (printer 1)

3) A card punch

4) If we execute a 999 halt at end-of-program, we want to conclude processing.

In order to create an environment with these characteristics, the following cards should be used:

```
                                    /-----------------------------------
                                    ! ?END
                                  /-----------------------------------
                                    ! FINI ;
                                /-----------------------------------
                                    ! EOJ = 999;
                              /-----------------------------------
                                    ! CARD.PUNCH ;
                            /-----------------------------------
                                    ! PRINTER1  POSITIONS = 132 ;
                          /-----------------------------------
                                    ! MEMORY.SIZE 9.6K ;
                        /-----------------------------------
                        ! ?DATA CARDS
                      /-----------------------------------
                      ! ?COMPILE B500/TEST WITH B500/IEP TO LI ;
                      !
```

Figure 4.3 Sample IEP Deck

EMULATOR I/O

In order to maintain consistency with MCP conventions and the philosophy behind them, all B500 peripherals are mapped onto existing file structures. In this fashion, MCP functions such as backup files, pseudo readers, and file equate capabilities are realized by emulators without significant problems.

Using file structures also enables the Interpreter to delay the binding of hardware until it is accessed, i.e., although an Interpreter may require two card readers; if they are never used at the same time during emulation, they may both be emulated by the same card reader. If a device is accessed while it is busy, either on the same Interpreter or on a native B1800/B1700 mode program, the Interpreter will wait for the FILE card before continuing and, therefore, eliminates the problem of run time contention for devices.

PPB INTERFACE TO THE INTERPRETER

The Run Structure is created by the MCP at BOJ time. Various fields or parameters are read from the PPB into the Run Structure. The Interpreter can thereby initialize the hardware scratchpad registers with data passed to it from the environment created by the B500/IEP.

S1A

B500.INDICATORS

FILLER	BIT(16)	Must be zero
SENSE.SWITCH	BIT(6)	May be set or reset
FILLER	BIT(2)	Must be zero

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

DESCRIPTOR INTERFACE

The PERIPHERAL.DESCRIPTOR is the major interface component between the B500 Interpreter and the B500 Interpreter Environment Program (IEP). The IEP is responsible for creating the 24-bit descriptor and passing it to the interpreter in the Program Parameter Block.

The PERIPHERAL.DESCRIPTOR defines the peripheral environment of the B500 described by the environment file in which it is contained and its format is shown below. A bit ON indicates that the corresponding device is in the environment; OFF indicates that it is not.

S1B

PERIPHERAL.DESCRIPTOR

	BIT # (from left)	
CARD.READER	0	
PRINTER1	1	
PRINTER2	2	
PUNCH	3	A File Parameter Block (FPB) for each of the devices referenced in the PERIPHERAL.DESCRIPTOR. They will exist in the name order, as well.
DISK.0	4	
DISK.1	5	
.	6	
.	7	
.	8	
.	9	
.	10	
.	11	
.	12	
DISK.9	13	
TAPE.1	14	
TAPE.2	15	
.	16	
.	17	
.	18	
TAPE.6	19	
LOAD.FILE	20	
<FREE>	21-23	

S2A

TARGET.MEMORY.BASE BIT(24)

S2B

STATIC.MEMORY.SIZE BIT(24)

S3A

EU.DENSITY.INDICATORS

DOUBLE.DENSITY.INDICATORS BIT(10)

---> TRUE = DOUBLE DENSITY

S4A

EOJ.9XX BIT(24)

---> Contains "9mn", where "mn" are two characters which represent the "m" and "n" variants of a halt description.

APPENDIX A: RAILROAD SYNTAX

Valid instructions for the B500 IEP program are presented in railroad-syntax diagrams, a method which allows visual representation of syntactic units. A B500 IEP instruction is constructed by following the main line of development from its beginning (>) to its point of termination (#). Where optional entries are concerned, the direction of flow proceeds down on the left-hand line and up on the right. Unless otherwise noted, optional entries are expected by the program in the order (top to bottom) in which they are presented.

Required entries occur on the main line of development; optional ones below the line. Required keywords appear in upper-case letters; variables in lower-case letters. Underlined portions of required keywords are acceptable abbreviations for the keywords. In the example of railroad syntax given in figure 6.1, the following conditions are implied:

* The main line of development is from left to right, down on the left vertical line and up on the right. A loop or series of choices is shown by a return line with an arrow showing a flow to the left (<----------).

* An arrow shows the direction of progress from the point of origination (>---) to termination (#). A statement is not complete until it reaches a terminator (#). Where the syntax continues on another line, a double arrow (>>) is used, both to break the first line and to continue the second. The continuation point is indented.

* Required syntax is presented on the main line of development; optional on a secondary line. Required keywords are presented in capital letters; variables in lower case. Underlined portions of keywords are their allowable abbreviations.

* The bridge /n\ shows the maximum number of times the line may be crossed. The bridge /n*\ shows that the line must be crossed at least once. Since required syntax is usually shown on the main line of development, the /n*\ option, presumably, will not need to be used often.

* Abbreviations may be used for variables. A list of standard abbreviations may be presented in the introduction of the document or the list may immediately precede the figure. In either case, the abbreviation(s)

should be as obvious as possible.

* Angle brackets (<>) and a space occur around variables; a space precedes and follows a required word.

* The pound sign (#) denotes a termination point for the syntax of the item being described.

* Continuation of a primary or secondary line of development can also be shown by a continuous line that returns (<--------) and then flows forward (---------->) to a vertical line. The return line may contain an entry. This format is helpful when an optional parameter contains several elements itself.

* A horizontal line between entries must be represented by at least a dash (--) in order to clearly distinguish the line from a hyphenated word.

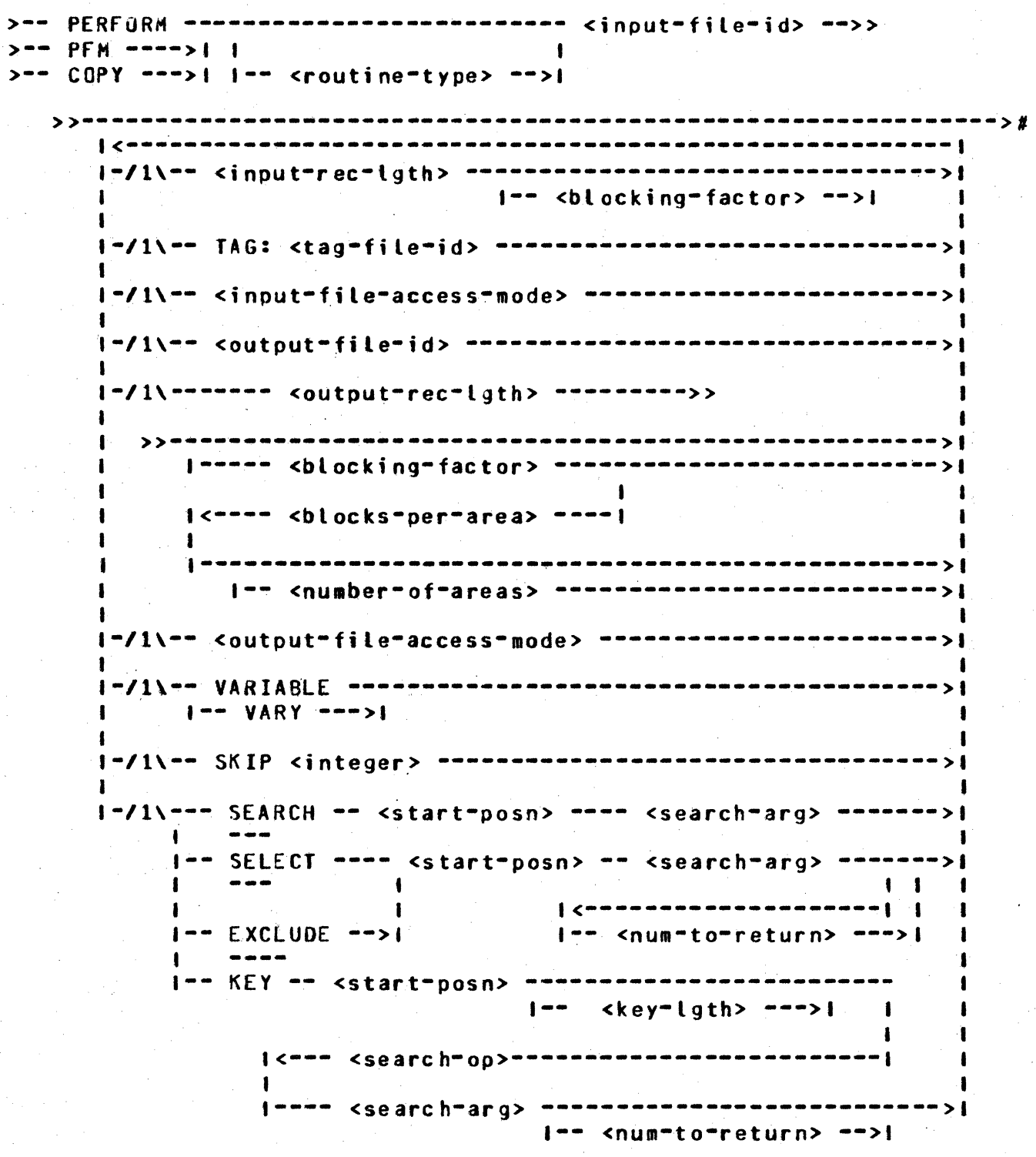


Figure 6.1 Sample of Railroad Syntax

BURROUGHS CORPORATION
COMPUTER SYSTEMS GROUP
SANTA BARBARA PLANT

INDEX