

Extended BASIC

System Manager's

Guide

093-000119-02

For the latest enhancements, cautions, documentation changes, and other information on this product, please see the Release Notice (085-series) supplied with the software.

Ordering No. 093-000119
©Data General Corporation, 1975, 1977
All Rights Reserved
Printed in the United States of America
Revision 02, October 1977
Licensed Material - Property of Data General Corporation

NOTICE

Data General Corporation (DGC) has prepared this manual for use by DGC personnel, licensees, and customers. The information contained herein is the property of DGC and shall not be reproduced in whole or in part without DGC prior written approval.

DGC reserves the right to make changes without notice in the specifications and materials contained herein and shall not be responsible for any damages (including consequential) caused by reliance on the materials presented, including but not limited to typographical, arithmetic, or listing errors.

Extended BASIC
System Manager's
Guide
093-000119

Revision History:

Original Release - February 1975
First Revision - June 1975
Second Revision - October 1977

093-000119-02 is a major revision of this manual. Therefore, change indicators have not been used.

The following are trademarks of Data General Corporation, Westboro, Massachusetts:

<u>U.S. Registered Trademarks</u>			<u>Trademarks</u>
CONTOUR I	NOVA	NOVALITE	DASHER
DATAPREP	NOVADISC	SUPERNOVA	INFOS
ECLIPSE			

Preface

Data General's Extended BASIC is a powerful, straight-forward language, which allows many different users to run their programs concurrently in a single computer installation. This kind of application - where different people share one system - often requires an administrator, or system manager. In many cases, the system manager is assisted by a system operator, who physically maintains the timesharing system.

This manual is intended for both system manager and operator. In most cases, the roles overlap somewhat; both manager and operator have privileged roles, and each must supervise the system for its users.

Chapter 1 of this manual outlines the range of hardware for Extended BASIC from a 24K single-user system to a 256K 32-terminal configuration. We also describe software applications, notation conventions, and symbols in this chapter.

Chapter 2 describes BASIC System Generation under the Real-Time Disk Operating System (RDOS) for RDOS applications; we explain each question of the BASIC SYSGEN dialog in order. We also explain how to load BASIC, and how to manage two important timesharing files (identification and accounting).

Chapter 3 describes daily operation and security precautions for the BASIC system, as well as the Extended BASIC privileged commands, which you can execute only from the system console.

Chapter 4 outlines the generation of Extended BASIC under the Stand-Alone Operating System (SOS).

The following Data General documents offer further information:

- 093-000003 *Binary Loader*
- 093-000075 *Real Time Disk Operating System Reference Manual*
- 093-000062 *Stand-Alone Operating System User's Manual*
- 093-000065 *Extended BASIC User's Manual*
- 093-000087 *BATCH User's Manual*
- 093-000109 *RDOS Command Line Interpreter Reference Manual*
- 093-000188 *How to Load and Generate Your RDOS System*

Documentation Conventions

We use these conventions for command formats in this manual:

COMMAND *required* [*optional*] ...

Where	Means
--------------	--------------

COMMAND	You must enter the command (or its accepted abbreviation) as shown.
----------------	---------------------------------------------------------------------

required	You must enter some argument (such as a filename). Sometimes, we use:
-----------------	-----------------------------------------------------------------------

$$\left\{ \begin{array}{l} \text{required}_1 \\ \text{required}_2 \end{array} \right\}$$

which means you must enter *one* of the arguments. Don't enter the braces; they only set off the choice.

[optional] You have the option of entering some argument. Don't enter the brackets; they only set off what's optional.

... You may repeat the preceding entry or entries. The explanation will tell you exactly what you may repeat.

Additionally, we use certain symbols in special ways:

Symbol	Character Represented	Explanation
)	Carriage Return	The RETURN key generates an automatic line feed in addition to the carriage return.
ESC	ESCApe or ALTmode key	In BASIC, the ESCape key echoes an asterisk (*) on your terminal.
□	Space	Sometimes used in this manual to emphasize a space character.
{ }	Braces	Indicate a choice of items enclosed.

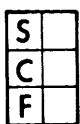
Terminal Devices

In this manual, the word "terminal" means teletypewriter, CRT display terminal, or equivalent interactive device.

Statements and Commands

The statements and commands described in Chapter 4 of this manual are presented as follows:

<table border="1"> <tr><td>S</td><td></td></tr> <tr><td>C</td><td></td></tr> <tr><td>F</td><td></td></tr> </table>	S		C		F		<h3>BASIC Word</h3> <p>Format:</p> <p>Purpose:</p> <p>Remarks:</p> <p>Examples:</p>
S							
C							
F							



A check in the box indicates whether the BASIC word is used as a statement (S), a command (C), or a function (F). You may use some BASIC words both as statements and commands.

Format:

The BASIC word is shown in its generalized format, according to the conventions described above.

Purpose:

A brief statement which describes the operation performed by the BASIC word.

Remarks:

This heading includes rules, cautions, and other pertinent comments on the BASIC word.

Examples:

We will provide some typical uses to help describe the BASIC word and its format.

Abbreviations

We will use the following abbreviations to describe BASIC statements and commands. The abbreviations are in boldface print in the formats and represent commonly used terms as follows:

Abbreviation	Meaning
var	numeric variable
expr	numeric expression
rel-expr	relational expression
str lit	string literal
val	numeric values
line no.	line number
col	column
control var	control variable
svar	string variable
mvar	matrix variable
filename	a disk filename or a device

We welcome your comments and suggestions for this and other Data General publications. To communicate with us, either use the comments form provided at the back of this manual, or write directly to:

Software Documentation
Data General Corporation
Westboro, MA 01581

End of Preface

Contents

Chapter 1 - Introduction to Basic System Management

System Hardware	1-1
System Software	1-1
Extended Memory Support	1-2
Overview of BASIC System Generation (BSG)	1-2

Chapter 2 - BASIC System Generation (BSG) Under RDOS for RDOS Object Systems

RDOS Sysgen	2-1
Loading BASIC Modules	2-1
BSG Dialog	2-1
Loading BASIC under RDOS	2-5
Disk Partitions and Directories	2-5
Creating BASIC Disk Directories and Files	2-6
Library Directory (BASIC.DR)	2-6
User Directories	2-6
BASIC.ID File	2-6
Log-On Programs	2-7
BASIC.AF File	2-7

Chapter 3 - System Operation Under RDOS

System Operator Responsibilities	3-1
Cold Starting BASIC under RDOS/MRDOS	3-1
System Preparation	3-1
Invoking the BASIC System	3-2
Device Independence	3-2
Communicating with Users	3-3
Changing Directories	3-3
System Console Breaks	3-3
Stopping the BASIC System	3-3
Suggested Shutdown Procedures	3-3
Recovery from System Crashes	3-4
Power Fail/Auto Restart Procedures	3-4

Chapter 3 - System Operator Under RDOS (continued)

Privileged Commands and Statements	3-4
ALL	3-5
BYE	3-5
CDIR	3-6
CPART	3-6
DIR	3-7
DISABLE	3-7
ENABLE	3-8
FALL	3-8
FMSG	3-9
FREE	3-9
GDIR	3-10
INIT	3-10
KILL	3-11
LEVEL	3-11
MAX	3-12
MSG	3-12
RELEASE	3-13
USERS	3-13
System Security Considerations	3-15
Controlling Access to the System	3-15
Directory Sharing	3-15
Monitoring the System	3-15
Protecting Programs and Data Files	3-15
System Backup Procedures	3-16
Batch Operation	3-16

Chapter 4 - Extended BASIC Operation Under SOS

Introduction	4-1
Starter System	4-1
Building SOS-Extended BASIC	4-1
Creating a Trigger	4-2
Performing a Relocatable Load (Paper Tape)	4-3
Performing a Relocatable Load (Magnetic Tape or Cassette)	4-5
Performing a Relocatable Load (RDOS Systems)	4-5
SOS BASIC SYSGEN Dialog	4-5
Sign-On	4-6
Restart	4-6
Power Fail/Auto Restart	4-7
Error Messages	4-7
Differences Between RDOS BASIC and SOS BASIC	4-8

Appendix A - Error Messages

Appendix B - ASCII Character Set

Appendix C - Privileged Command Summary

Appendix D - Sample BSG Dialog

Chapter 1

Introduction to Basic System Management

System Hardware

We have designed the Extended BASIC software system for use on any Data General computer. You can generate and run a small, single-user BASIC system with the following minimum hardware configuration:

- Data General computer with at least 24K words of main memory
- DGC 6052/6053 display or other interactive ASCII terminal, and a high-speed paper tape reader.

You can build a large, full-scale interactive time-sharing BASIC system by expanding the minimum configuration to include any combination of the following items:

- Up to 256K words of main memory
- Hardware multiply/divide
- Hardware floating-point unit
- Multiline asynchronous interfaces
 - ALM8 8-line asynchronous multiplexor, or
 - ALM16 16-line asynchronous multiplexor, or
 - 4060 32-line asynchronous multiplexor (QTY)
- 2 Moving-head disk controllers (8 drives)
- 2 Fixed-head disks
- 2 Magnetic tape controllers (16 transports)

- 2 NOVA cassette controllers (16 transports)
- 2 High-speed paper tape readers
- 2 High-speed paper tape punches
- 2 Card readers
- 2 Line printers

NOTE: You can add a second system console (DGC 6052/6053 display or other interactive ASCII terminal) to run in the foreground under RDOS/MRDOS for dual operations. Therefore BASIC can run with applications such as RJE or SORT/MERGE.

System Software

Data General's BASIC can run under the Real Time Disk Operating System (RDOS) and Mapped RDOS (MRDOS).

Mapped RDOS supports configurations requiring more than 32K words of main memory.

RDOS is upward compatible with MRDOS as summarized in Table 1-1.

All unmapped RDOS features except BATCH operation and the use of cassettes are supported by DOS.

If you wish to build and run BASIC under Data General's Stand-Alone Operating System (SOS), see Chapter 4.

Table 1-1. BASIC Supported Devices and Hardware Features

System Feature	RDOS	MRDOS
Minimum Memory	24K	32K
Maximum Memory	32K	256K
Hardware Multiply/ Divide	Yes	Yes
Floating-Point Processor	Yes	Yes
ALM8 Asynchronous Multiplexor	Yes	Yes
ALM16 Asynchronous Multiplexor	Yes	Yes
4060 Asynchronous Multiplexor (QTY)	Yes	Yes
Cassette transports	0-16	0-16
Magnetic Tape transports	0-16	0-16
Moving-head disks	0-8	0-8
Fixed-head disks	0-2	0-2
Card readers	0-2	0-2
Line printers	0-2	0-2
High-speed paper tape readers	0-2	0-2
High-speed paper tape punches	0-2	0-2
Second console	Yes	Yes

Extended Memory Support

In an MRDOS system, you may use storage above 31K words in the ground in which BASIC is running for a high-speed swapping area. To accomplish this, use the CLI SMEM command to set the size of the ground. For example, in an MRDOS system which has 128K words of memory, you could organize the memory allocations as follows:

operating system	MRDOS 32K
background	BASIC 31K
	HIGH-SPEED SWAPPING AREA 34K
foreground	FORTRAN 31K

In this system, FORTRAN requires 31K in the foreground, and BASIC requires 31K in the background. The CLI command

SMEM 65)

would provide an additional 34K in the background for BASIC to use as a high-speed swapping area.

Overview of BASIC System Generation (BSG)

BASIC system generation requires a host system (either RDOS or MRDOS) to run the BSG program. The questions in the BSG dialog allow the system manager to select a configuration for a BASIC system which, in turn, will run under an RDOS or MRDOS object system.

Chapter 2 of this manual describes BASIC system generation, Chapter 3 covers operation and maintenance of the BASIC system, and Chapter 4 explains how to generate and operate a SOS BASIC system.

End of Chapter

Chapter 2

BASIC System Generation (BSG) Under RDOS for RDOS Object Systems

RDOS SYSGEN

Some BASIC features require RDOS features, and cannot be used unless you include these features when you generate RDOS. For example, a multiuser BASIC requires either an ALM or QTY multiplexor driver in the operating system.

Also, if you want to use the ESCape character as your interrupt character, you must change the default interrupt character to include it. Respond with "27" (ESC) for the first interrupt character and "128" (none) for the second interrupt character to the appropriate RDOS SYSGEN question.

The RDOS system generation program, SYSGEN.SV, or BSYSGEN.SV, configures a system save file based on the characteristics you name during the system generation dialog. Many of the questions in the RDOS SYSGEN dialog relate to BASIC requirements; the appropriate answers can tailor RDOS for BASIC.

You will find the complete RDOS SYSGEN dialog and instructions in the manual *How to Load and Generate Your RDOS System*.

Loading BASIC Modules

BASIC is supplied as a group of library modules and relocatable binary modules corresponding to your computer configuration. You may transfer modules to disk by the RDOS CLI command XFER or LOAD, as applicable.

BSG Dialog

To configure the BASIC system, enter the RDOS CLI command:

BSG)

This command invokes the BSG program, which asks you about your desired configuration for Extended BASIC.

After you enter the command BSG), you will see the following identification message and instructions:

EXTENDED BASIC REVISION X.X MM/DD/YY SYSTEM
GENERATION

where:

X.X is the revision level of this BASIC and
MM/DD/YY is the release date.

VALID ANSWERS ARE IN PARENTHESES. DEFAULT VALUES ARE ALWAYS GIVEN FIRST AND MAY BE SELECTED BY A NULL RESPONSE (CARRIAGE RETURN). DEFAULTS ARE FOLLOWED BY A LIST OF ALTERNATIVES, ANY ONE OF WHICH MAY BE SELECTED BY TYPING A MATCHING RESPONSE.

The following queries and their applicable responses during BSG are numbered in this manual for your convenience. No such numbering occurs during BSG. You must terminate each response (except that to question 20) by a carriage return.

1. OBJECT COMPUTER TYPE (NOVA,ECLIPSE, NOVA3,ORIGINAL) ?

Respond by naming the object computer type for the BASIC system. ORIGINAL refers to the original NOVA processor.

2. OBJECT OPERATING SYSTEM (RDOS,RTOS,DOS) ?

Respond with the type of operating system under which the BASIC system will run.

3. SYSTEM SAVE FILE (BASIC, ANY DISK FILE NAME) ?

Specify the filename of the system you are generating. This may be BASIC or any other filename. The filename specified will name system files, e.g.:

name.SV (executable core image file)
name.OL (overlay file)
name.CL (command line file)
name.RB (relocatable binary file)
name.DL (copy of system generation dialog for system being configured)
name.MP (load map file)

4. ACCOUNTING (YES,NO) ?

Answer YES if you want the system to create file BASIC.AF, to record accounting information.

5. MULTI-USER SYSTEM (NO,YES) ?

Respond with NO if the system you are configuring will be a single-user system. Respond YES for a multiuser system. If you respond NO, all queries in the BSG dialog which relate to multiuser systems are skipped. To respond YES, you must have ALM or QTY support in your RDOS system.

6. LINE CONFIGURATION:

(Asked only if the answer to question 5 is YES.) Respond with a list of line numbers, a range of line numbers, or both. BASIC supports lines 0-31 (maximum of 32 lines). List line numbers separated by commas, or give a range of line numbers; for example:

0,1,2,4 means line numbers 0, 1, 2, and 4
0-2,4 means line numbers 0, 1, 2, and 4

7. MASTER CONSOLE LINE NUMBER (-1,0-31) ?

(Asked only for a multiuser system.) The master console is usually the same terminal which the RDOS CLI uses for its communication, but in certain extraordinary cases, you may find it necessary to assign it to a multiplexor line. This line number must be within the range you specified in question 6. If you choose this option, you cannot use the CLI console (\$TTI/\$TTO in BG; \$TTI1/\$TT01 in FG) as a BASIC user terminal.

8. IS MASTER CONSOLE A CRT (YES,NO) ?

BSG wants to know whether the RUBOUT key echo sequence should be a backarrow (←) for hard copy terminals or an erase-last-character for cathode ray terminals (such as the DGC 6052/6053 terminal).

9. CENTRAL PROCESSOR OPTION SUPPORT:

(heading for queries 9a through 9e).

NOTE: If you specify a hardware feature that does not exist on your object machine, the system will generate no specific error message.

- 9a. DIAL-UP LINES (NO,ALM,4060) ?

(Asked only if answer to question 5 is YES). Select NO if you are not using dial-up lines; select ALM or 4060, as appropriate, if you are using dial-up lines. (RDOS has its own support for the ALM; BASIC provides support for the 4060.)

- 9b. SWAPPING (YES,NO) ?

(Asked only if you're configuring BASIC as a multiuser system.) Answer NO if you do not desire disk swapping, or YES if you want swapping. The swapping file, BASIC.SW, will be built in the default directory.

- 9c. HARDWARE MULTIPLY/DIVIDE (NO,YES) ?

(Asked only if you're using a NOVA computer. This feature is standard on ECLIPSE computers.) Indicate whether or not your computer is equipped with hardware multiply/divide.

- 9d. HARDWARE FLOATING POINT (NO,SING,DOUB) ?

If hardware floating point is included in your system, specify whether you will run it as a single- or double-precision unit.

A NO response will result in a single-precision, software-driven system.

Licensed Material - Property of Data General Corporation

9e. MEMORY MANAGEMENT/PROTECTION (NO, YES) ?

(This question is skipped for RDOS systems with original NOVA computers.) Indicate whether your machine is equipped with this optional feature.

10. USER WRITTEN SUBROUTINES (NO,YES) ?

Indicate whether or not you will include user-written subroutines in your BASIC system. If your answer is YES, then a relocatable binary file named SBRTB.RB, constructed in accordance with Appendix B of the *Extended BASIC User's Manual*, must reside in the current directory.

11. DEFAULT FEATURES ACCEPTABLE (YES,NO) ?

(See question 12.)

12. DEFAULT OVERLAYS ACCEPTABLE (YES,NO) ?

(Asked only if your answer to question 11 is YES. If your answers to questions 11 and 12 are both YES, the system will include the default selection for questions 13a through 13w without printing the questions at your terminal. If your answer to either question is NO, then BSG will prompt you to respond to each question.) The use of overlays increases the amount of user storage available, but slows down the system when you use these features.

You must determine which features to include as overlays or in main storage to provide an efficient BASIC system. The overlay area requires 512 words.

13. VALID ANSWERS TO THE FOLLOWING ARE:

O=INCLUDE FEATURE AS AN OVERLAY
M=INCLUDE FEATURE IN MAIN MEMORY
NO=DO NOT INCLUDE FEATURE

13a. MATRIX ARITHMETIC (O,M,NO) ?

Includes: Matrix assignment, addition and subtraction, zero matrix (ZER), multiplication (MUL), unit matrix (CON), determinant (DET), identity matrix (IDN), transposition (TRN).

13b. MATRIX INPUT AND OUTPUT (O,M,NO) ?

Includes: MAT READ FILE
MAT INPUT [FILE]
MAT PRINT [FILE]
MAT WRITE FILE

13c. MATRIX INVERSE (O,M,NO) ?

Matrix inversion (INV).

13d. PRINT USING (O,M,NO) ?

Includes PRINT USING logic.

13e. BYE (O,M) ?

13f. CALL (O,M) ?

Question 13f is asked only if your system will include user subroutines.

13g. MARK SENSE CARD READER (O,M,NO) ?

If you have a card reader, and plan to use abbreviated format DGC mark-sense cards, answer O or M. Otherwise, answer NO.

13h. LIST,PUNCH (O,M) ?

13i. FILE COMMANDS (O,M,NO) ?

13j. USERS COMMAND (O,M) ?

13k. OPERATOR COMMANDS (O,M) ?

Includes: DISABLE, ENABLE, FREE, KILL, MAX, MSG, ALL, FALL, FMSG, LEVEL

13l. NEW (O,M) ?

13m. USER INITIALIZATION (O,M) ?

Called each time a user logs on.

13n. DELETE,RENAME,PAGE,RENUMBER,SIZE, INIT,RELEASE (O,M,NO) ?

13o. CONSOLE CHARACTERISTICS (O,M,NO) ?

13p. DISK AND DIRECTORY COMMANDS (O,M,NO) ?

13q. STRING ARITHMETIC (O,M,NO) ?

13r. ERROR MESSAGE TEXT (O,M,NO) ?

13s. I/O ERROR MESSAGE TEXT (O,M,NO) ?

13t. SIN,COS,TAN,ATN (O,M,NO) ?

13u. LOG,EXP,↑,SQR,RND (O,M,NO) ?

13v. LEN,POS,STR\$,VAL (M,NO) ?

13w. RUN FILENAME,CHAIN,SAVE,LOAD,LOGON (O,M,NO) ?

(LOGON is the auto log-on feature.)

14. MAXIMUM FILES TO BE OPEN AT ONE TIME (N,1-M) ?

The values printed for N and M in this query will vary according to your responses to previous queries. BASIC determines the values for N and M as follows:

M = either A or B, whichever is smaller, where

A = 9 times the number of users, and
B = 255-Q, where

Q = 3 for all BASIC systems (including [S]BASIC.SD (background) or [S]F\$BASIC.SD (foreground)), plus 2 channels for all multiuser systems, plus 1 for swapping (BASIC.SW), plus 1 for accounting (BASIC.AF), plus 2 channels for each user terminal (including iden\$. \$\$ where iden is the abbreviated account ID).

N = either C or D, whichever is larger, where

C = 9, and
D = 2 times the number of users

NOTE: You must consider such BASIC file commands as ENTER, LOAD, SAVE and LIST in the total, since they reference files.

15. FILE SHARING (YES, NO) ?

This question refers to the file sharing feature of BASIC by which active users can access other accounts' files, subject to restrictions. Each user may open as many as eight shared or unshared files at one time subject to the pool restriction described in question 14 above.

16. FOR-NEXT NESTING LIMIT (8,4-20) ?

Increasing the depth of nesting consumes main memory by six words per level, single-precision, and ten words per level, double-precision.

17. GOSUB NESTING LIMIT (8,4-20) ?

Increasing the depth of nesting consumes main memory by one word per level.

18. USER FUNCTION NESTING LIMIT (6,0-10) ?

Increasing the depth of nesting consumes main memory by three words per level, single-precision, and five words per level, double-precision. If you specify 0, then you may not use any user functions in your program.

19. MAXIMUM USER CORE SIZE (IN BYTES) ?

This is an extremely useful optimization feature. We recommend that you use it not only to decrease the amount of swapping that might occur but also to allow swapping to overlap with useful processes.

Recommended procedure for utilizing this feature:

- 1) Generate a test BASIC system without core size limit. Note the result of your SIZE command.
- 2) Generate another BASIC system that limits core size so that at least one-fourth of all active users can fit in main storage at the same time. This will not only limit the number of pages that must be swapped to the disk but will also allow swapping overlap in process.
- 3) If you receive complaints from users that they haven't enough room, you can slowly increase the core size, keeping in mind the effect on performance noted above.

20. PROMPT STRING (*-SPACE-NULL, ANY 10 CHARACTER STRING) ?

Select the default prompt (*) or choose your own prompt string of up to 10 characters.

Every prompt string must end with one of the following valid delimiters: null (CTRL-SHIFT-P), carriage return, or form feed. This delimiter is included as the last character permitted in the prompt string.

21. TIME OF DAY WITH PROMPT (NO,YES) ?

Answer YES if you want the time of day output with each prompt.

Question 21 is the last question in the BSG dialog.

Loading BASIC Under RDOS

After you have answered the queries of the BSG dialog, three files are created: name.CL, name.RB, and name.DL.

NOTE: "name" is the filename given in response to BSG question number 3.

Name.RB is the BASIC system configuration module.

Name.CL contains a relocatable load command line of the modules which correspond to your responses to the various BSG queries.

Name.DL is a copy of the BSG dialog and your responses for the system created.

RDOS automatically loads the BASIC system after the last BSG query. It loads only those modules which are specified in the name.CL file.

It is not necessary to invoke BSG to reload a mirror image of the .SV file; rather, execute the name.CL indirectly (e.g., @BASIC.CL@). This command line file will create name.SV, name.OL and name.MP.

Disk Partitions and Directories

RDOS permits the parceling of disk file space among several users, on both a fixed and a variable basis. (See Figure 2-1.) Fixed parcels of a contiguous disk file space are called *secondary partitions*. Secondary partitions are mutually exclusive subsets of total disk file space, the *primary partition*.

You can allocate file space within any partition on a variable basis. That is, you may allocate mutually exclusive portions of a partition, and these portions may shrink or grow within the limits of total file space available in the parent partition. These variable parcels of file space are called *subdirectories*.

Refer to the *RDOS Reference Manual*, and to the *Extended BASIC User's Manual* for a complete description of files, directories and disk partitions.

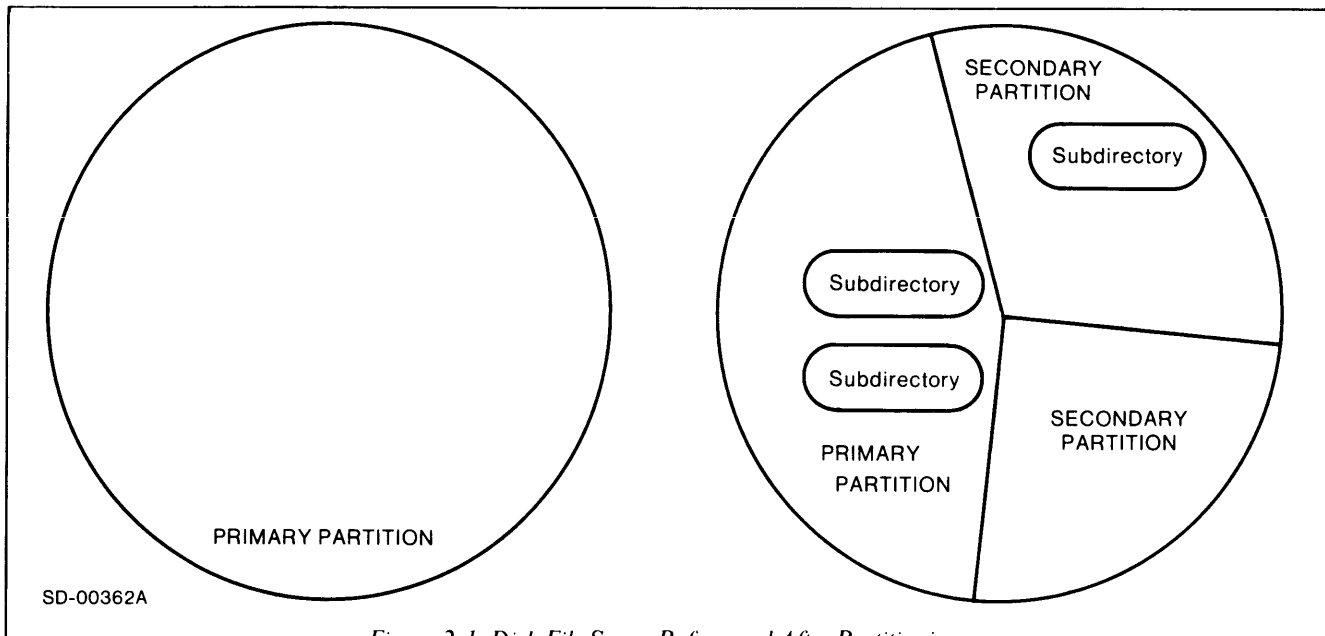


Figure 2-1. Disk File Space Before and After Partitioning

As the system manager, you must parcel the disk file space available in the BASIC system. Since a system might include more than one moving-head disk, you should assign one disk (e.g., DP0) as the master device. You should INIT and DIR this device so that it becomes the default directory; and you may create all directories on other devices using the RDOS CLI commands.

For example:

INIT DP0)	Initialize DP0
DIR DP0)	Make DP0 the default directory
CDIR SUB1)	Create a subdirectory on DP0
CPART HASA 50)	Create a secondary partition with 50 blocks

The commands (or statements) CDIR, CPART, and INIT are available under BASIC as well as RDOS to create and initialize subdirectories and secondary partitions. You may execute these commands only from your system console; they are not available to users. For more detail, see the Privileged Commands section of Chapter 3.

Generally, you should allocate fixed parcels of disk space (secondary partitions) to users of a timesharing system while keeping the BASIC system library in a subdirectory.

Creating BASIC Disk Directories and Files

Before invoking the BASIC system, you must create certain directories and files, as described next.

Library Directory (BASIC.DR)

The BASIC system library directory (BASIC.DR) must always reside on the master device as either a secondary partition or a subdirectory.

You create the library directory by a CDIR BASIC or CPART BASIC command. This directory can include frequently used BASIC utility programs and user-donated programs. User log-on programs are SAVED in BASIC.DR.

From the system console, you SAVE or LIST programs into BASIC.DR. These programs can then be LOADED, ENTERED, or RUN from any user's terminal and, in that respect, can be shared by all users.

User Directories

The secondary partitions or subdirectories that you assign to each user are called user directories. They provide storage space for the user's programs and data files. You must create each user directory in the system with a CDIR or CPART command.

You can allow shared user directories by specifying the same directory for each sharing user in the BASIC.ID file.

BASIC.ID File

On multiuser Extended BASIC systems operating under RDOS, an ASCII disk file must be present to contain the account identification of all system users. You can create and modify the BASIC.ID file by using a BASIC program or by using one of the utility text editors provided with RDOS.

Each account identification entry that you make in the BASIC.ID file must fit in a single line, and must be terminated by a carriage return. The format for each line is as follows:

iden/ [password] [[dev:] dir [filename])

iden: The user account identification (ID) (range 4-20 characters). The characters must be alphanumeric (0 to 9 and A to Z only) and each *iden* entry in the file must be unique.

password: an optional four-character (alphanumeric) password.

dev: required if directory is not on master device (for example, DP0).

dir: the optional name of the secondary partition or subdirectory assigned to the account as the user's directory.

It is considered bad practice to assign more than one user to a directory since it defeats BASIC's file security.

filename: the name of an optional log-on program SAVED in the library directory (BASIC.DR). Any log-on program included in the user's account identification entry will automatically be executed after a successful log-on. See *Log-on Programs* for further information.

The contents of a BASIC.ID file might appear as follows:

```
THOM/SUB1/PROF
1111/KAST/JERR
JONE//HASA/HELLO
0002/PROG/USER2
0003//USER3/HELLO
```

Licensed Material - Property of Data General Corporation

Log-On Programs

There are several reasons to include an optional log-on program with the user identification in the BASIC.ID file.

1. A user wants a specific program to be executed each time he logs on to the system.
2. You, the system manager, want to have a general program executed each time a user logs on.
3. A combination of the above, where you want to run a general program with the user's specific program.

You must SAVE the log-on program in the library directory, BASIC.DR, and include the program name with the user's identification entry in file BASIC.ID. Log on programs are executed immediately after a successful log-on by the user.

To prevent the interruption of a log-on program, BASIC always disables the recognition of the ESC key just before program execution begins. Therefore, you should include a CHAR statement enabling escapes as the first line of your log-on program or at the end of the BASIC.DR log-on program.

The following common log-on program, originated by the system manager, provides some flexibility for both the system manager and the user. The program requires that you name all log-on programs specified in the BASIC.ID file HELLO.SV, and that each user's log-on program be SAVED in his/her respective directory. The common log-on program must also be named HELLO.SV and it must be SAVED in the library directory (BASIC.DR). Therefore, you can alter the common log-on program, and each user can alter his/her own log-on program, as necessary.

The user's program should include the CHAR statement as the first line of the program, to cancel the NOESC executed when the common log-on program is run.

```

0005 REM- COMMON LOGON PROGRAM
0008 REM- (HELLO.SV)
0010 REM- IN SYSTEM DIRECTORY (BASIC.DR)
0015 IF SYS(11) < 12 THEN GOTO 0040
0020 IF SYS(11) < 18 THEN GOTO 0050
0025 GOTO 0060
0040 PRINT "GOOD MORNING"
0045 GOTO 0065
0050 PRINT "GOOD AFTERNOON"
0055 GOTO 0065
0060 PRINT "GOOD EVENING"
0065 IF CPU(0) THEN GOTO 0085
0070 PRINT "SYSTEM GOING DOWN FOR";
0072 PRINT "□ MAINTENANCE"
0075 PRINT "COME BACK AT 14:30"
0080 BYE
0085 REM- CHAIN TO HELLO.SV IN USER'S
0086 REM- DIRECTORY
0090 CHAIN "HELLO.SV"
0095 END

```

BASIC.AF File

If your response to BSG question 4, "ACCOUNTING (YES,NO) ?", is YES, the system creates a **direct file** named BASIC.AF (unless it already exists). *pa. dir.*

The BASIC.AF file resides in the primary partition of the master device. All information which the system writes to BASIC.AF is appended to the end of the file. The accounting file will continue to grow as BASIC log-ons or BYE's are executed, and you should process this file periodically to keep from using up too much disk space.

The system writes each entry to BASIC.AF as a one-line string of ASCII characters. Four lines are written for each user. Any of the lines for various system users may be interleaved with lines for other system users. However, each information line has a precise meaning and uniquely identifies a user. Therefore, it is easy to recover fully the accounting information stored by writing a BASIC program which RENAMES this file and then opens a (new) file called BASIC.AF. Since each line in the file is pure ASCII text, it is easy for a BASIC program to process the data. Because there is a space delimiter between each field, string functions can readily access the subentries.

In this way, you can periodically remove old accounting information for processing, even while BASIC is running.

The format for lines in the BASIC.AF file is:

iden mm/dd/yy HH:MM SIGN ON, NN
iden mm/dd/yy HH:MM SIGN OFF, NN
iden mm/dd/yy HH:MM CPU USED, t
iden mm/dd/yy HH:MM I/O USED, s, s1

where:

iden user ID (4 alphanumeric characters): OPER for the system console

mm month (e.g., 09)

dd day (e.g., 23)

yy year (e.g., 77)

- HH hour of day (24-hour clock)
- MM minute
- NN port number (e.g., 03; -1 for system console)
- t integer number of seconds of CPU time used since log-on for this port.
- s integer which relates to the number of BASIC file I/O calls (e.g., OPEN FILE, READ FILE, WRITE FILE, etc.)
- s1 integer which is the total BASIC I/O statements (ENTER, SAVE, DISK, WHATS, etc.)

An example of a segment of a BASIC.AF file follows:

```
OPER 09/23/75 12:20 SIGN ON, -1
0000 09/23/75 12:21 SIGN ON, 01
0001 09/23/75 12:21 SIGN ON, 00
0000 09/23/75 12:22 SIGN OFF, 01
0001 09/23/75 12:22 SIGN OFF, 00
0000 09/23/75 12:23 CPU USED, 15
0001 09/23/75 12:23 CPU USED, 4
0000 09/23/75 12:23 I/O USED, 1,10
0001 09/23/75 12:23 I/O USED, 0,0
0000 09/23/75 12:23 SIGN ON, 01
0001 09/23/75 12:24 SIGN ON, 00
0001 09/23/75 12:27 SIGN OFF, 00
0000 09/23/75 12:27 SIGN OFF, 01
0001 09/23/75 12:27 CPU USED, 31
0001 09/23/75 12:27 I/O USED, 1, 12
0000 09/23/75 12:27 CPU USED, 45
0000 09/23/75 12:27 I/O USED, 3, 13
OPER 09/23/75 12:27 SIGN OFF, -1
OPER 09/23/75 12:27 CPU USED, 0
OPER 09/23/75 12:27 I/O USED, 0, 14
```

End of Chapter

Chapter 3

System Operation Under RDOS

In a typical timesharing BASIC system, both the system manager and the system operator have privileged roles; each must supervise vital parts of the system for its users. This chapter addresses the needs of both operator and manager, including maintenance, privileged commands, and security measures.

System Operator Responsibilities

The system operator for a timesharing system generally has the responsibility for controlling and maintaining the system's resources on a real-time basis.

The operator's duties may range from the mounting and dismounting of magnetic tapes and the distribution of line printer output, to the updating of user identification to control system access by potential users.

The BASIC system provides one console with special capabilities and commands which cannot be executed on any other console connected to the system. This privileged console is called the system console (SC); it is usually the same system console used by RDOS. The system console offers all the normal features of Extended BASIC, and can be used as if it were any other user terminal.

In a background/foreground system, two system consoles are available; each is the SC for its own BASIC system. When running BASIC in both background and foreground simultaneously, you may use only one ground for a multiuser system, because the ALM and QTY cannot be shared between grounds.

Cold Starting BASIC Under RDOS/MRDOS

A cold start is defined as the system start-up procedure for use when the CPU has been halted or turned off in an orderly manner, or when you're bringing the system up after an initial RDOS SYSGEN and BASIC SYSGEN (as described in Chapter 2). The restart procedure follows an inadvertent crash. Both cold start and restart procedures are described later in this chapter.

System Preparation

1. Turn on the system console, the CPU, and the disk unit you're using to boot the system.
2. The disk mounted in the disk unit must have a copy of the BOOT program and a properly generated RDOS/MRDOS operating system save file on it as described in the manual, *How to Load and Generate Your RDOS System*.
3. Load the RDOS/MRDOS systems by performing the appropriate disk bootstrap procedure (described in *How to Load and Generate Your RDOS System*.)
4. Proceed using either an existing BASIC system save file, or generate a BASIC system as described in Chapter 2. Note that more than one BASIC system save file may be available.

Invoking the BASIC System

Use the DIR command to access the directory which contains the BASIC systemname.SV file and type the name of this file in response to the CLI prompt.

For example:

```
R
SMALLBASIC)
```

where SMALLBASIC.SV is located in the current directory, and was the name given to the BASIC system save file.

Once BASIC is running it will output the message:

```
BASIC REVISION X.X MM/DD/YY
```

where X.X is the revision number of this version and MM/DD/YY is the date on which system testing and development of this version was completed.

At this point BASIC performs a number of initialization tasks before it proceeds. These tasks include initializing the library directory and creating a swapping file, if necessary. If an initialization error occurs, an error message is output to the system console; BASIC will stop and exit to the CLI. Initialization error messages are listed and described with corrective actions in Appendix A. Correct any errors before reinvoking BASIC.

The system is now ready for use provided the BASIC.ID file, the BASIC common library directory (BASIC.DR), and user directories have been created as described in Chapter 2.

Device Independence

In a timesharing environment, there is often more than one of each type of secondary storage and I/O device in the system (e.g., magnetic tape and line printers). And, more often, it is impractical to provide each user with a permanently reserved device.

Therefore, the resources of the timesharing system must be shared in some practical manner which minimizes inconvenience to users and promotes efficient use of system peripherals.

One such technique includes the concept of device independence, where the users' programs are written to defer the actual naming of a peripheral device until program execution time. Typically, this is accomplished in a program as follows:

```
.
.
.
100 INPUT "TAPE UNIT ",T$
110 LET F$ = T$, ":5"
120 OPEN FILE (3,2),F$
.
.
.
```

Before this program is run, the user sends a message to the system operator, requesting that a specific tape be mounted. This message should include any requirements for removing the write-enable ring. The system operator then acknowledges receipt of the message, locates the user's tape, mounts it on an available magnetic tape drive, INITIALizes the device, and returns a message to the user stating which drive was selected.

The dialog at the system console might go something like this:

```
*FROM JERR: MOUNT TAPE #D033 WRITE ENABLE.
```

```
*MSG JERR MSG RCVD.) ←
*INIT "MT5") ←
*MSG JERR D033 ON MT5. GO AHEAD.) ←
```

```
Operator acknowledges message. _____
Mount and INIT the device. _____
Allow user to proceed. _____
```

Now that you have specified MT5, the user can proceed as follows:

```
*ENTER PROG)
*RUN)

TAPE UNIT: MT5) (query from program)
.
.
.
```

After the programming session, the user notifies the system operator to dismount the magnetic tape, and thereby free the tape transport for another user. Dialog at the system console might be:

```
*FROM JERR: DONE. REMOVE #D033 FROM MT5.
*MSG JERR MSG RCVD. HAVE A NICE DAY. )
*RELEASE "MT5")
```

Communicating with Users

You can execute four commands from the system console to communicate with active users:

MSG Message to single user
 FMSG Forced Message (overrides NOMSG)
 ALL Message to all users
 FALL Forced message to all users (overrides NOMSG)

We'll describe each of these commands later in this chapter.

Changing Directories

You can change the current directory with the DIR command. This ability to change directories at the system console allows the system operator to maintain files in any directory in the system, even if the user assigned to the directory is logged on. The system operator can access any file in the directory unless a user has already opened it for I/O.

You can change directories at the system console without interrupting normal operation or disturbing other users in a multiuser environment.

System Console Breaks

There are three RDOS system break characters that can be input from the system console:

CTRL-A (background and foreground)-interrupt
 CTRL-C (background and foreground)-interrupt and save
 CTRL-F (background only)-foreground interrupt from background

These break characters can abort the BASIC system, and are described in the *RDOS Command Line Interpreter Reference Manual*.

When operating BASIC, the CTRL-A and CTRL-C system breaks serve no practical purpose, since BASIC is equipped with an ESC break and a BYE command to return to the CLI. The use of CTRL-A, CTRL-C or CTRL-F under BASIC can result in the loss of accounting information in the BASIC.AF file and incomplete or erroneous data files.

To prevent the inadvertent use of CTRL-A and CTRL-C, use the BASIC commands DISABLE and ENABLE (described under Privileged Commands).

Note that system break characters (except for ESC) from users' terminals have no meaning to BASIC and are ignored.

Stopping the BASIC System

Occasionally it may be necessary to shut down BASIC for maintenance or to regenerate a system with a different configuration. An orderly shutdown procedure will help minimize inconvenience to users.

Suggested Shutdown Procedures

Steps	Comments
1. *MAX=1)	Prevent any new user from logging on.
2. *ALL SYSTEM COMING DOWN IN 5 MIN PLS) *ALL LOGOFF BY THEN. THANKS.)	
3. *USERS)	Determine who's still signed on.
NAME STATUS... 0000 0042... OPER 0042...	} User 0000 still signed on.
4. *FMSG 0000 PLS LOGOFF)	Send the reluctant user another warning message.
5. *BYE) ERROR 64 - SYSTEM ACTIVE	Note: Operator can't kill an active system.
6. *USERS) NAME STATUS... 0000 0042 OPER 0042...	
7. *KILL 0000)	Force user 0000 off the system.
8. *BYE)	Return control to RDOS.
OPER 11/13/75 17:29 SIGN OFF, -1 OPER 11/13/75 17:29 CPU USED,0 OPER 11/13/75 17:29 I/O USED,0,0 17:29:10	} System manager's accounting information displayed.
R	

Recovery from System Crashes

If your system inadvertently fails, you should document the circumstances of the failure before proceeding to re-invoke BASIC.

- A. Record the contents of the ADDRESS lights.
- B. Record the contents of each of the four accumulators (AC0, AC1, AC2, and AC3).
- C. Document as completely as possible which peripherals, directories, and users were in use or on line at the time of the crash.
- D. Re-BOOT the RDOS system.
- E. CLEAR all files in the primary partition and other directories which were in use at the time of the crash. (CLEAR/A/V/D).
- F. Re-invoke the BASIC system.
- G. Send appropriate messages (MSG or ALL) after the affected users log on again. Note that the BASIC.AF file will not show log-offs for users on the system at the time of the crash.

Power Fail/Auto Restart Procedures

RDOS provides software support for the power fail/automatic restart option.

When power is restored, if the console key is in the LOCK position, you will see the message

POWER RESTORED

on the system console. If the console key was in the ON position when input power failed, set the console switches to all zeros (down) and press START when power is restored. This outputs the console message and restores the BASIC system as when the key is in the LOCK position.

The following system devices are given power-up restart service:

- paper tape reader/punches
- Teletypes
- quad multiplexors
- card readers
- line printers
- disks

Character devices may lose one or more characters during power up. Each card reader may lose up to 80 columns of information on a single card. Line printers may lose up to a single line of information. Since power-up service for disks includes a complete re-read or rewrite of the current disk block, no disk information is lost, although moving head disk units will require 30 to 40 seconds before disk operations can continue. Devices requiring operator intervention (like line printers, card readers, etc.) must receive such action if power was lost for an extended period of time.

No power-up service is provided for magnetic tape or cassette units.

Privileged Commands and Statements

You can execute most of the following BASIC commands (statements) only from the system console. The commands MSG, GDIR, and BYE are not privileged commands, but are discussed from the system operator's point of view.

Several commands are BASIC implementations of RDOS commands; some (e.g., CDIR, CPART) are functionally identical to their RDOS versions, although their syntax differs. The commands appear in alphabetical order.

The format used to describe these commands is discussed in Chapter 1.



ALL

Format:

ALL message

Purpose:

To transmit a message to all active users who have not set NOMSG.

Remarks:

1. All active users on the system who have not set NOMSG will receive the transmitted message on their terminals. The message would appear as follows:

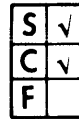
```
FROM OPER:message
```

where OPER is the ID associated with the system console.

2. No error message will appear at the system console if a user has NOMSG set.
3. Message length is limited to one line per ALL command.
4. Quotation marks are not necessary for message.

Example:

```
*ALL LINE PRINTER OUTPUT IS READY)
*
```



BYE

Format:

BYE

Purpose:

To return to the operating system CLI.

Remarks:

1. The BYE command causes an exit to the operating system (RDOS) CLI unless one or more users are still logged on.

In this case, you will see the error message:

```
ERROR 64 - SYSTEM ACTIVE
```

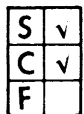
on the system console and you will not be able to exit from BASIC.

You should inform (FALL command) all active users that the system is coming down, then try the BYE command again. It may be necessary to KILL (KILL user ID) a reluctant user after fair warning.

2. A successful BYE command adds the system operator's accounting information to the accounting file (BASIC.AF) and displays the accounting information on the system console.

Example:

```
*BYE)
OPER 10/28/75 09:27 SIGN OFF, -1
OPER 10/28/75 09:27 CPU USED, 158
OPER 10/28/75 09:27 I/O USED, 3023, 1272
R
```



CDIR

Format:

CDIR name

name: a string literal or string variable of up to 10 characters.

Purpose:

To create a subdirectory named name.DR

Remarks:

1. Each user must be assigned to a subdirectory (or to a secondary partition).
2. You may create the subdirectory in the master directory, or in any other primary or secondary partition, provided you have LINKed it from the master directory via the RDOS CLI.
3. Several users may share the same subdirectory. (See Chapter 2 for more information.)

Examples:

```
*CDIR "SMITH")  
*
```

Creates SMITH.DR in current directory.

```
*CDIR "KAST")  
*
```

Creates KAST.DR.



CPART

Format:

CPART name,size

name: a string literal or string variable of up to 10 characters.

size: number of contiguous blocks.

Purpose:

To create a secondary partition named name.DR

Remarks:

1. Each user must be assigned to a secondary partition (or to a subdirectory).
2. You may create the secondary partition in the master directory, or in any other primary partition.
3. Several users may share the same secondary partition. (See Chapter 2 for more information.)
4. If size is not an integer multiple of 16, then the system truncates size to the next lower multiple.

Examples:

```
*CPART "JOHN", 192)  
*
```

Creates secondary partition named JOHN.DR on current directory.

```
*CPART "DON", 256)  
*
```

Creates secondary partition named DON.DR.

S	√
C	√
F	

DIR

Format:

DIR { primary part [:secondary part][:subdirectory] }
{ secondary part[:subdirectory] }

Purpose:

To change the current directory on the current primary device to another directory on the same or different primary device.

Remarks:

1. If you do not specify the primary partition, then the directory specified in the command is assumed to be in the current primary partition.
2. If necessary, the DIR command will initialize the device or directory specified in the command.
3. This command is an implementation of the RDOS CLI DIR command.

Example:

```
*DIR DP1:DEF:GHI)
*
```

Change the current directory to subdirectory GHI in secondary partition DEF in primary partition DP1.

S	√
C	√
F	

DISABLE

Format:

DISABLE

Purpose:

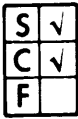
To prevent the inadvertent use of the CTRL-A, CTRL-C and CTRL-F (background) RDOS system console breaks.

Remarks:

1. If you accidentally type one of the RDOS system console breaks, BASIC will abnormally terminate. The DISABLE command issued at the BASIC master console prevents the recognition of the three control characters at the RDOS system console.
2. You can cancel the DISABLE command with an ENABLE command.
3. A DISABLE command executed while BASIC is running in the foreground will inhibit CTRL-F on the background console.

Example:

```
*DISABLE)
*
```



ENABLE

Format:

ENABLE

Purpose:

To cancel a DISABLE command.

Remarks:

This command restores recognition of the CTRL-A and CTRL-C and CTRL-F (background) RDOS console breaks.

Example:

```
*ENABLE)  
*
```



FALL

Format:

FALL message

message: text of message

Purpose:

Forces the transmission of a message to all active users.

Remarks:

1. This command overrides the NOMSG command set by active users.
2. When you use this command, the following is printed at each user's terminal:

FROM OPER message
3. Message length is limited to one line per FALL command.
4. Quotation marks are not necessary for message.

Example:

```
*FALL ***SYSTEM GOING DOWN IN 5 MINUTES***)  
*FALL ***PLS LOG OFF WITHIN 5 MINUTES***)  
*
```




FMSG

Format:

FMSG userID message

userID: identification of receiving user.

message: text of message.

Purpose:

Forces the transmission of a message to a specific user.

Remarks:

1. This command will override a NOMSG command if set by the receiving user.
2. When you use this command, the following is printed at the user's terminal:

FROM OPER:message

where OPER is the ID associated with the system console.

3. Message length is limited to one line per FMSG command.
4. Quotation marks are not necessary for message.

Example:

```
*FMSG JACK PLEASE LOG OFF.)
*
```

On Jack's terminal, this message will appear:

FROM OPER: PLEASE LOG OFF.



FREE

Format:

FREE userID

userID: a four-character user account identification (ID).

Purpose:

To interrupt execution of the program which is being processed for userID.

Remarks:

1. Normally, the user can interrupt program execution by pressing the ESCape key. However, if the user's program is locked in an inescapable loop, assistance from the system operator may be necessary.
2. When the FREE command is executed, the message UNLOCKED BY OPERATOR and an asterisk (*) prompt is output to the affected user's terminal.

Example:

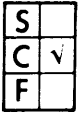
```
*10 ON ESC GOTO 20
*20 ON ERR GOTO 30
*30 X=1/0
40 GOTO 10
*RUN
```

} User program (account JOHN) in inESCapable program loop.

```
*FREE JOHN) Command at system console
*
```

The following message would be output at JOHN's terminal:

```
UNLOCKED BY OPERATOR
*
```



GDIR

Format:

GDIR

Purpose:

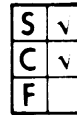
To print the name of the current directory.

Remarks:

1. The command is useful at the system console, particularly when you use the DIR command frequently to change directories.
2. Users may also give the command, but it has limited use: a user can access only the directory assigned to his/her user identification.
3. This command is an implementation of the RDOS CLIGDIR command.

Example:

```
*GDIR)
BASIC2 All file references are currently to directory
      BASIC2.
*
```



INIT

Format:

INIT { svar
 "str lit" }

svar and str lit: the name of a directory or device expressed as either a string variable or a string literal.

Purpose:

To initialize a directory or a device and thereby permit access to its files.

Remarks:

1. In BASIC, you will normally use INIT to initialize reserved file name devices, such as magnetic tape units and cassette units. You must INIT and RELEASE these devices for each user.

Examples:

```
*INIT "MT0")
*INIT "CT3")
*B$="DP7")
*INIT B$)
*INIT "DC3")
```

S	v
C	v
F	

KILL

Format:

KILL userID

userID: a four-character user account identification (ID).

Purpose:

To force a specific user off the system.

Remarks:

This command will force userID off the system. Accounting information will be displayed on the user's terminal as follows:

```
LOGGED OFF BY OPERATOR
NAME MM/DD/YY HH:MM SIGN OFF, XX
NAME MM/DD/YY HH:MM CPU USED, YY
NAME MM/DD/YY HH:MM I/O USED, ZZ, QQ
```

Example:

*KILL JACK)

This information will be displayed on JACK's terminal:

```
LOGGED OFF BY OPERATOR
JACK 12/16/75 17:29 SIGN OFF 03
JACK 12/16/75 17:29 CPU USED 8
JACK 12/16/75 17:29 I/O USED 3, 12
```

S	v
C	v
F	

LEVEL

Format:

LEVEL [userID] [n]

userID: a four-character user account identification (ID).

n: a priority level constant in the range $0 \leq n \leq 50$.

Purpose:

To monitor or set the priority constant for any user, including the system operator.

Remarks:

1. The default value of n is 25.
2. The LEVEL command can either raise or lower the priority level (n) of any user's tasks. The lower the value of n, the higher the priority.
3. For equal priority levels, tasks are handled on a first-in, first-out basis.

Examples:

1. *LEVEL)

The current value of the priority level constant (n) for the system operator is printed on the system console. Any user can issue this form of the command to determine his/her own value of n.

2. *LEVEL JOEL)

This prints the value of the priority level constant for user JOEL on the system console.

3. *LEVEL JOEL 10)

This sets the value for the priority level constant (n) for user JOEL to 10.

4. *LEVEL 30)

This sets the system console to a lower than default priority.

S	v
C	v
F	

MAX

Format:

MAX [= val]

val: a number from 0 to 33.

Purpose:

To establish a limit for the number of active users.

Remarks:

1. The initial value for MAX is 33.
2. You may set MAX to a value smaller than the current number of active users.
3. If the number of active users exceeds the current MAX value, then a user attempting to log on will receive the following message and be denied access to the system:

MAXIMUM USERS

4. A user can log on successfully if the number of active users is less than the current value of MAX.
5. If you give the MAX command without the val argument, the the system will return the current value of MAX.

Examples:

```
*MAX = 7)
*
*MAX)
7
*
```

S	v
C	v
F	

MSG

Format:

MSG [userID message]

userID: identification of receiving user.

message: text of message.

Purpose:

To transmit a message to a specific user on the system.

Remarks:

1. If the receiving user has set the message lockout command (NOMSG) or is not on line, then the transmission will not succeed and BASIC will print the error message:

ERROR 52 - USER IN NOMSG STATE

on the system console.

2. If the transmission succeeds, then BASIC will print the following at the receiving user's terminal:

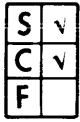
FROM OPER:message

where OPER is the ID of the system console.

3. Message length is limited to one line per MSG command.
4. Quotation marks are not necessary for message.
5. When used without operands, the MSG command resets the action of a NOMSG command.
6. Generally, it is not advisable to use a NOMSG at the system console, since timesharing users cannot override NOMSG.

Example:

```
*MSG JACK YOUR TAPE IS ON MT1:--GO AHEAD)
*
```



RELEASE

Format:

RELEASE { svar }
 { "str lit" }

svar and str lit: the name of a directory or a device expressed as either a string variable or a string literal.

Purpose:

To prevent further I/O access to a previously INITed directory or device.

Remarks:

1. You use the BASIC version of the RDOS command primarily to RELEASE magnetic tape and cassette units. When you release these devices, the system automatically rewinds tapes mounted on them.
2. You may also RELEASE directories and disks; you must release a disk before removing any disk pack.
3. RELEASE partitions and subdirectories in the order in which they are nested.
4. You need not RELEASE line printers or card readers.

Examples:

```
*DIR "DP1:JOE:BILL")
*RELEASE "BILL")
*RELEASE "JOE")
*RELEASE "DP1")
*
```



USERS

Format:

USERS [userID]

userID: user account identification.

Purpose:

To print a status report, on the system console, of all active users. When you include the optional userID, BASIC prints the status of that user only.

Remarks:

The status report contains the following information:

Heading	Meaning
NAME	Four-character user identification where OPER is system operator's identification.

STATUS	Four hexadecimal digits. Four zeros mean an idle terminal.
--------	------------------------------------------------------------

Hexadecimal Value	Meaning
1	Run equals on, edit equals off
2	CLI console doing [MAT] INPUT (doesn't echo carriage return)
4	Swap in progress or swapped out
8	Program storage has been changed
10	Don't allow messages
20	Execute-only file is in use
40	Active user (successful sign-on)
80	BYE in progress
400	Timed input in progress
800	No subdirectory associated with this user
1000	Terminal input in progress
2000	Delay in progress
8000	Special mark-sense card translation

For example, 004C means:

0040	active user
0008	program storage changed
0004	swap in progress or swapped out
<u>0001</u>	run mode

004C

USERS (continued)

Heading	Meaning	Heading	Meaning
DIRECTORY	The name of the directory assigned to NAME, from BASIC.ID file.	PROG	Two numbers. The first number is the core block address of the program; the second number is the number of program blocks in use for this user. An ellipsis (...) indicates the program has been swapped out of main memory.
LINE	QTY or ALM line number (0 to 31). System console is line -1.	PRI	The current priority level set by LEVEL.
SWAPS	The number of accesses to file BASIC.SW made for this user.	OVLY	The number assigned to the current overlay in the queue or execute area. (-1 means no overlay in use.)
DATA	Two numbers. The first number is the core block address of the data; the second number is the number of data blocks in use for this user. An ellipsis (...) indicates the data has been swapped out of main memory.	FILES	Number of files opened at this time by this user.
		CPU	Number of seconds of CPU time used by this user.

Example:

*USERS)

NAME	STATUS	DIRECTORY	LINE	SWAPS	DATA		PROG		PRI	OVLY	FILES	CPU
0000	0041	USER0	0	9	8	32	40	4	25	-1	0	3.7
TREK	104C	TREK	1	1	...	4	...	4	25	-1	1	.1
JERR	0041	JERRY	2	7	44	32	76	4	25	-1	0	1.8
OPER	0044	BASIC	3	5	...	32	...	4	25	9	0	11.8
	0000		4									

System Security Considerations

This section of the manual describes some precautions that you, as system manager, can take to make your BASIC system secure. You must not only maintain the BASIC timesharing system, but also safeguard the security and integrity of user programs and data files. Data General's BASIC provides the capabilities by which you can:

- restrict access to the system by any potential user;
- control directory sharing;
- monitor and control the use of system facilities; and
- restrict access to program and data files.

Controlling Access to the System

The BASIC.ID identification file controls all access to the BASIC System. This file contains the four-character identification code, password, directory assignment, and log-on program name for each system user. The file is described in Chapter 2. The entries which you make to this file provide a complete, exclusive set of system users.

No one can use the system without knowing a user identification code listed in the BASIC.ID file, and only one user at a time may use that code. You can create, delete, or modify entries in this file as often as necessary.

As each user logs on, BASIC outputs the ACCOUNT NAME: prompt at his terminal.

The four-character user identification code is not necessarily secure information, since it is required as an argument to the MSG communication command, appears in any listing of the BASIC.AF accounting file, and is displayed on the system console in response to a USERS command. Therefore, BASIC allows you to append an optional password to each user entry in the BASIC.ID file. This password is a four-character extension to the user identification code. The user and system manager select it, and it appears only in the BASIC.ID file.

After you enter a user's password in the BASIC.ID file, BASIC will issue a PASSWORD: prompt whenever that user logs on. The password itself is never echoed on the terminal. Three incorrect answers to the prompt will place that user's terminal back in the ready state, where it will await another ESCape.

If you (or a user) suspect that an unauthorized person has learned a password, you should change the BASIC.ID file promptly.

You should keep the BASIC.ID file itself secure from examination or modification by anyone else. You should not keep this file in any directory accessible to system users (such as BASIC.DR), nor should you ever CHATR it to have the "O" sharable attribute. You could CHATR it to be write-protected ("W") and permanent ("P"), although you would have to remove these attributes whenever you modified the file.

Directory Sharing

Although you must assign a unique identification code to each user, several users may share the same directory or partition. However, approach directory sharing with extreme caution for the following reasons:

- If two users share a directory, one may freely use the other's files, and can inadvertently destroy them by using the same filename in a LIST or SAVE command.
- Data files can be inadvertently overwritten in the same way.

Monitoring the System

In addition to the cumulative record of system usage maintained in the accounting file (BASIC.AF), you can get an immediate report on active users with the USERS command. With this report and communication commands MSG, FMSG, ALL, and FALL, you can determine user activity and obtain desired information from a user.

If an unauthorized person is using someone else's log-on identification, there are several options. The MSG and FMSG commands enable you to establish a dialog with the offending user and convey any instructions or ultimatum desired. The FREE and KILL commands can halt that user's program or force the user off the system, respectively. After the offending user has logged off, you can change the now-unsecure identification code in the BASIC.ID file.

Protecting Programs and Data Files

Normally, users have complete access to all the files in their own directories, and limited access to files in public directory (BASIC.DR) and other user's directories. You can access any users' directory from the system console by naming that directory in a DIR command.

The BASIC system can make any user's files accessible by other users on a selective basis, or restrict the type of access even for the owner of a file. The CHATR command provides this capability. Attributes which the BASIC CHATR command can set are compatible with those of the RDOS CHATR command.

You can examine the attributes of any file by using the BASIC WHATS command, or the RDOS CLI LIST/E command. File attributes of particular importance in maintaining file security are:

- P Permanent file. File cannot be deleted unless this attribute is removed.
- R Read protected. File cannot be opened for reading.
- W Write protected. File cannot be opened for writing.
- O Sharable file. File may be accessed by other users when they know the name of the directory in which the file resides, and the name of the file.
- E Execute only. You may LOAD or ENTER and RUN a program, but you cannot LIST, SAVE or modify it.
- H Special BASIC argument to CHATR which specifies attributes O, P and W.

File attributes apply to all users, including the owner, and can be changed only by the owner or the system manager. It may be necessary to temporarily change the attributes of a file before it can be used. For example, a user might intentionally specify attributes "PW", to prevent the inadvertent deletion or overwriting of a file. The "W" attribute would have to be removed with the CHATR command before the file could be changed.

For one user to access a file in another user's directory, the following conditions must be met:

1. The user must know the name of the appropriate directory and file.
2. The file must have the "O" sharable attribute.
3. The mode of access to the file must not conflict with any other attributes set for the file.
4. The number of files currently being shared in the system must not exceed the maximum specified during BASIC system generation.

System Backup Procedures

We recommend highly that you follow a periodic procedure for preserving system and user programs. Each procedure will vary from one installation to the next, depending on hardware configurations and user requirements.

Generally, you should shut down the system during nonprime hours and transfer the contents of each directory in the system to magnetic tape or a disk

Licensed Material - Property of Data General Corporation

cartridge set aside specifically for backup. You can make the transfer at the RDOS CLI level using the DUMP command. You should store tapes or disks used for backup in a secure area away from the immediate location of the computer.

Batch Operation

BATCH is an RDOS program which allows system software to manage the processing of a job stream without operator intervention. When Extended BASIC runs under BATCH, the BASIC system becomes a single-user system dedicated to BATCH at the system console. This is true even for multiuser configurations. Any attempt to log on while the system is running under BATCH evokes the message:

BATCH

If the RDOS system is configured for background/foreground, then a multiuser BASIC and a BASIC under BATCH may run concurrently if there is no multiplexor conflict.

The BATCH commands begin with an exclamation point, as in the BATCH command to execute BASIC:

R (RDOS CLI prompt)
BATCH) (execute BATCH)
!JOB ID:xxxx (xxxx is the card which contains the first BATCH command in the job; it might include the user ID.)
!name (where name is the name of the BASIC system save file (.SV) specified during BSG.)

When using BASIC under BATCH, the user writes the same series of commands and statements into the jobstream as would be input from a terminal.

BATCH requires an input file, an output file, and a log file. The default assignment for the log file is the console printer; the default device for the input file (called SYSIN) is the card reader (\$CDR); and the default device for the output file (called SYSOUT) is the line printer (\$LPT). These default files may be changed by the user, as described in the *BATCH User's Manual*.

Extended BASIC commands under BATCH must not attempt any input/output to unit record devices assigned to SYSIN or SYSOUT. For example, the following are *illegal* commands if \$CDR is SYSIN or \$LPT is SYSOUT:

CARDS "\$CDR")
LIST "\$LPT")

Licensed Material - Property of Data General Corporation

If your system is configured for a mark-sense card reader, and you plan to use DGC abbreviated format cards, you must include the /T switch with the BASIC system save filename as follows:

```
R
BATCH)
!JOB ID:xxxx
!name/T      (where name is the name of the BASIC
              system save file.)
.
.
!EOF
```

The /T switch indicates that each command or statement in the SYSIN (\$CDR) jobstream after you invoke BASIC is in abbreviated format and will be translated by BASIC.

If SYSIN is not a card reader, and you specify a CARDS "\$CDR" (with abbreviated format cards), then you do not need the /T to specify translation for the cards. However, the SYSIN file must be in standard BASIC format. CARDS "\$CDR" is the only BASIC statement which will recognize the DGC abbreviated format mark-sense cards when you have not set the /T switch.

You can use the mark-sense card reader as SYSIN (using abbreviated format cards and specifying the /T switch) and you can execute a CARDS command or statement for a disk file which is not in abbreviated format. The /T refers only to card translation.

Note that a mark-sense card reader does not necessarily require abbreviated format cards; you can use conventional cards in it as well. The /T switch is not required with conventional cards.

In the BSG dialog, the query

MARK SENSE CARD READER (O,M,NO) ?

determines whether or not the card reader will be used for abbreviated format cards. If it is, specify Overlay (O) or Main (M); otherwise, specify NO.

The example of BATCH BASIC in Figure 3-1 uses a mark-sense card reader as SYSIN. Note that the commands in the jobstream (\$CDR) are prepared on standard cards in Hollerith code. When you execute BASIC/T from the jobstream, all subsequent commands and statements are prepared on cards in the DGC abbreviated format.

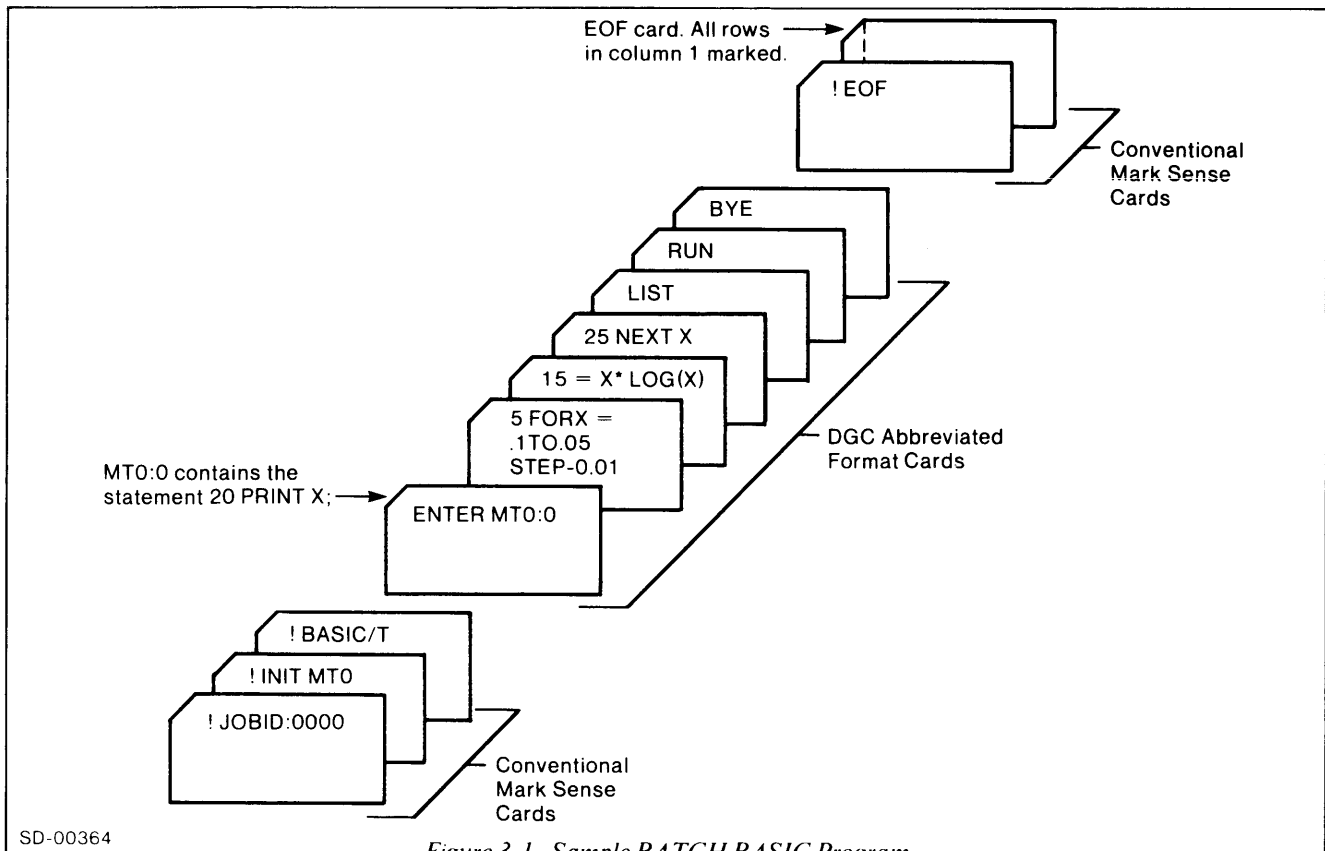


Figure 3-1. Sample BATCH BASIC Program

End of Chapter

Chapter 4

Extended BASIC Operation Under SOS

Introduction

A stand-alone version of Extended BASIC is available for use on systems without a direct access I/O device. This chapter describes the generation and operation of BASIC under such a system. Although stand-alone systems differ only slightly from their RDOS BASIC counterparts, we have included a list of differences at the end of this chapter.

Before building a SOS BASIC system, you must know the concepts and operation of the Stand-Alone Operating System (SOS), as described in the *Stand-Alone Operating System User's Manual*.

Starter System

A small starter system, capable of running in 12K words of core storage, is available. This system is configured at the factory and contains the following features:

- A device driver for the system console
- A device driver for an 80-column line printer
- A device driver for a high-speed paper tape reader
- A device driver for a high-speed paper tape punch
- PRINT USING capability
- Matrix manipulation capability

You can load the absolute binary (core image) paper tape for this system with the Binary Loader (091-000004), as described in the Binary Loader manual. When loading is complete, the system will halt. Press the CONTINUE switch, enter the current date and time, and your system will be ready to accept input.

Building SOS-Extended BASIC

For systems with at least 16K words of storage, you may tailor SOS-Extended BASIC to support a number of different hardware configurations. You must perform the following steps to configure your system:

1. Produce a trigger which specifies the desired I/O support and program features.
2. Perform a relocatable load of the trigger, the appropriate SOS libraries, and the BASIC libraries and relocatable binaries.
3. Perform a run-time system generation to further tailor the system.

Creating a Trigger

The SOS SYSGEN program produces a trigger during system generation. SYSGEN accepts a command line which contains device driver ENTRY symbols from the console device. It outputs a relocatable binary file (the trigger) of EXTERNAL NORMAL symbols corresponding to the named device drivers. These EXTERNAL NORMALS select or "trigger" the desired routines for loading when they precede the SOS libraries as input to the relocatable loader.

To create a trigger, load and start the SYSGEN program. If your SYSGEN program is on paper tape, load it with the Binary Loader (091-000004). You can load SYSGEN from cassette or magnetic tape with the Core Image Loader/Writer. You can also build SOS BASIC on an RDOS system as described in the Performing a Relocatable Load (RDOS Systems) paragraph below. Only one of the following SYSGEN tapes will be supplied with your system:

Tape No.	Description
091-000070	SYSGEN without mass storage
091-000071	SYSGEN with cassette (paper tape for systems with cassette capability)
091-000074	SYSGEN with magnetic tape (paper tape for systems with magnetic tape capability)

When the SYSGEN program is started, it outputs the prompt:

SYSG

and waits for a command line response. The command line has the format:

SYSG driver₁ ...driver_n .RDSI output-device/O BTRIG/T)

where:

driver₁ ... driver_n is a list of entry symbols in the desired device driver routines. Table 4-1 lists all possible symbols.

output-device is the name of the device to which the trigger is to be output. This name must be followed by the "O" switch.

BTRIG/T BTRIG is the title which the /T switch assigns to the trigger.

Table 4-1. Driver Entry Symbols for SYSGEN Command Line

Driver Entry Symbol	SOS Program Name	Function
.CDRD	CDRDR	card reader driver (includes mark-sense)
.CTAD	CTADR	cassette driver for unit 0
.CTU1	CTU1	control table/buffer for cassette units 0-1
.CTU2	CTU2	for cassette units 0-2
.CTU3	CTU3	units 0-3
.CTU4	CTU4	units 0-4
.CTU5	CTU5	units 0-5
.CTU6	CTU6	units 0-6
.CTU7	CTU7	for units 0-7
.L132	LP132	132-column line printer driver
.LPTD	LPTDR	80-column line printer driver
.MTAD	MTADR	magnetic tape driver for unit 0
.MTU1	MTU1	control table/buffer for magnetic tape units 0-1
.MTU2	MTU2	for mag tape units 0-2
.MTU3	MTU3	units 0-3
.MTU4	MTU4	units 0-4
.MTU5	MTU5	units 0-5
.MTU6	MTU6	units 0-6
.MTU7	MTU7	units 0-7
.PTPD	PTPDR	high-speed paper tape punch driver
.PTRD	PTRDR	high-speed paper tape reader driver
.TTI1	TTY1	second console Teletype (TTI1, TTO1)

Licensed Material - Property of Data General Corporation

For example, to produce a trigger for the following devices:

- paper tape reader
- paper tape punch
- 132-column line printer
- mark-sense card reader

you would respond to the SYSG prompt with

```
.PTRD .PTPD .L132 .CDRD .RDSI $PTP/O BTRIG/T)
```

**Performing a Relocatable Load
(Paper Tape)**

Before you can load your system, you will need to create an absolute binary tape from your trigger, your SOS Library tape, and other tape modules you want in your system. Other modules include the BASIC library, PRINT USING routines, Matrix function routines, and so on (see Table 4-2).

You will also need the Binary Loader, the Extended Relocatable Loader and the SOS Relocatable Binary Punch Tapes. A list of these tapes, in order of loading, follows:

Tape No.	Description
091-000004	Binary Loader (ignore if already in core from SYSGEN)
091-000038	Extended Relocatable Loader
user-built	Trigger
099-000041	SOS Cassette Library (for Cassette support)
099-000042	SOS Magnetic Tape Library (for Mag Tape support)
099-000010	SOS Common Library
089-000080	SOS Relocatable Binary Punch

1. Using the Binary Loader, load the Extended Relocatable Loader.
2. Mount the trigger in the Teletype reader and type 1 or if using the high-speed reader, type 2.

3. Set the switch register to 1000, type 3.
4. Mount the TTYDR.RB tape in the Teletype reader and type 1, or mount it in the high-speed reader and type 2.
5. If magnetic tape or cassette support is included in the trigger, you must load the appropriate magnetic tape library or cassette library before the SOS common library. Mount the SOS Library in the Teletype reader and type 1, or in the high-speed reader, type 2.
6. Mount the relocatable binaries and libraries in the order shown in Table 4-2 on the Teletype reader and type 1, or on the high-speed reader, type 2.
7. Type 5 and note the value of NMAX output by the relocatable loader on the Teletype; you will use this number in step 13.
8. Mount the relocatable binary punch program on the Teletype reader and type 1, or on the high-speed paper tape reader and type 2.
9. Type 6 and note the value of RBFP output by the relocatable loader on the Teletype; you will use this number in step 11.
10. Type 8 to terminate the loading process.
11. Enter RBFP (from step 9) into the data switches on the computer console, press RESET then START.
12. Type 0H for output on the Teletype punch, or 1H for output on the high-speed punch.
13. Type 1, nmaxP where nmax is the value of NMAX noted in step 7.
14. Type 377E, to specify a starting address for the program.
15. Load the paper tape output from this procedure with the Binary Loader.
16. Proceed to the SOS BASIC SYSGEN dialog.

Table 4-2. Extended BASIC Supplied Paper Tapes

Tape Name	Supplied As	Purpose	Remarks
MP.RB MP26.RB	089-000137 089-000140	Driver for system console Driver for system console and 4026 multiplexor	Choose one
MP60.RB	089-000141	Driver for system console and 4060 multiplexor	
MDSW.RB	089-000156	Standard multiply/divide routines (all machines)	Choose one
MHDW.RB	089-000157	Multiply/divide routines for machines having options 8007, 8107, 8207, 8307	
MDNO.RB	089-000158	Multiply/divide routines for machines having option 4031	
MSCR.RB	089-000162	Translator for mark-sense cards (card number: N96830)	Optional
BASICA.LB	099-000046	Compiler routines	Required
BASIC.LB	099-000047	Interpreter routines	
SBRTB.RB	user-written	CALL resolutions (see Extended BASIC User's Manual, Appendix B)	Optional
MAT.RB	089-000138	Matrix function routines	
PRU.RB	089-000139	PRINT USING routines	
BASIC7.LB	099-000048	Single-user library	Choose one
BASIC8.LB	099-000049	Multiuser library	

Performing a Relocatable Load (Magnetic Tape or Cassette)

If your SOS system has a magnetic tape or cassette, you can create a BASIC save file by using the SOS CLI command RLDR (see *SOS User's Manual*, Chapter 3). The order of input of the relocatable binaries is the same as for paper tapes. The keysheet supplied with each tape lists the tape file number for each binary.

Performing a Relocatable Load (RDOS Systems)

You can build your entire BASIC system on an RDOS disk-based system, and subsequently transfer it to your SOS system by following this procedure:

1. Create a trigger source file from the console:

```
XFER/A $TTI BTRIG.SR)

.TITLE BTRIG
.COMM TASK,0
.EXTN .RDSI
.EXTN .PTRD
.
.
.
.END

|Z
```

} :SOS device drivers from Table 4-1
(Read in through Teletype reader.)

2. Assemble the trigger:

```
MAC BTRIG)
```

3. Transfer the relocatable binaries from tape to disk:

```
XFER $PTR SOS.LB)
```

or

```
XFER MT0:0 SOS.LB)
.
.
.
} other modules for your
} BASIC system
```

4. Perform a relocatable load:

```
RLDR/Z/N $LPT/L BASIC/S BTRIG 1000/N|)
TTYDR SOSMT.LB SOSCT.LB SOS.LB|)
MP60 MSCR MDSW BASICA.LB BASICB.LB|)
MAT PRU BASIC8.LB)
```

5. Test the resultant system:

```
BOOT BASIC)
```

NOTE: You must re-initialize RDOS after this test.

6. Make a core image file for loading on the machine without a disk:

```
MKABS/Z/S BASIC $PTP)
```

SOS BASIC SYSGEN Dialog

The BASIC system generation process, called SYSGEN, configures your system by asking questions and noting your answers. It is possible to configure several different BASIC systems and to save each one on paper or magnetic tape.

The core image file of all the modules in your BASIC system is on paper tape. To invoke SYSGEN, load the tape with the Binary Loader. BASIC starts automatically and identifies itself (you can restart SYSGEN at any time by pressing the ESC key):

```
BASIC REVISION X.X MM/DD/YY
```

where:

X.X represents the revision level. You should note this number on all correspondence with DGC.

MM/DD/YY represents the date that system testing at the factory was completed.

At this point, a single error message may occur:

```
INCOMPATIBLE OPERATING SYSTEM
```

signifying that the SOS.LB is Revision 8 or earlier. If you get this error message, contact your local DG representative and tell him/her that you want SOS revision 9.

Having identified itself, BASIC proceeds with the SYSGEN dialog. If you do not have a multiuser system, the next two questions are skipped.

LINE CONFIGURATION: (multiuser systems only)

Respond with a list of terminal line numbers, a range of terminal line numbers, or both. Terminate your answer with a carriage return. The 4026 multiplexor can address 16 lines (0-15) and the 4060 multiplexor can have a maximum of 32 lines (0-31). The maximum number of working terminals which Extended BASIC can service is 33 (32 multiplexor lines and a system console). Separate each line number by a comma and specify ranges by a dash:

```
0,1,2,4          lines 0, 1, 2 and 4
3-7, 10, 12-13, 17 lines 3, 4, 5, 6, 7, 10, 12, 13 and 17
0-31             lines 0 through 31 inclusive
```

DIAL-UP LINE CONFIGURATION: (multiuser systems only)

If any of the terminals you specified in the first question are to be on dial-up lines, enter their numbers. A carriage return specifies no dial-up lines.

RESERVED FILENAMES:

You must respond with a list of reserved filenames taken from Table 4-3. You must include in the trigger, BTRIG.RB, the appropriate driver for any device in the list for later access. Terminate the list with a carriage return.

Note that \$TTI and \$TTO are not in the list. These files may be accessed by the ENTER, LIST, PUNCH, PRINT and INPUT keyboard commands.

Table 4-3. Reserved Filenames

Reserved Filename	Device
SCDR	[mark-sense] card reader
CT0: CT1: CT2: CT3: CT4: CT5: CT6: CT7:	cassette unit 0 cassette unit 1 cassette unit 2 cassette unit 3 cassette unit 4 cassette unit 5 cassette unit 6 cassette unit 7
SLPT	line printer
MT0: MT1: MT2: MT3: MT4: MT5: MT6: MT7:	magnetic tape unit 0 magnetic tape unit 1 magnetic tape unit 2 magnetic tape unit 3 magnetic tape unit 4 magnetic tape unit 5 magnetic tape unit 6 magnetic tape unit 7
SPLT	plotter
SPTP SPTR	paper tape punch paper tape reader

After you enter your reserved filenames, BASIC asks:

ERROR MESSAGE TEXT?

Respond with a Y (YES), carriage return (YES) or anything else (NO). By default, BASIC reports each error as a two-digit code, followed by a brief description of the error. If you respond NO, the system will output only the two-digit code on each error.

BASIC SYSGEN is now complete and the computer will halt. You can save the entire configured system by invoking the Relocatable Binary Punch program, RBFP. Locate the symbol RBFP on the load map, enter the address opposite the RBFP symbol into the data switches, and press RESET and START. If no save is desired, press CONTINUE.

BASIC now asks for the date, and you respond:

DATE: MM-DD-YY)

and for the time:

TIME: HH:MM)

in 24-hour notation.

Sign-On

The system console is now activated:

10/01/75 15:33 SIGN-ON, SC

BASIC attempts to allocate storage for each user. If there is not enough core for each user, the error message

NO CORE

is output and the system will halt. You must then either generate a SOS with fewer features or fewer users, or a BASIC with fewer features, or both.

You can activate the multiplexor terminals by pressing the ESC key once:

10/01/75 15:34 SIGN-ON, 0
*

Restart

You may stop the system at any time and restart it at location 377. At restart, BASIC automatically clears each user's working storage area with the NEW command.

Power Fail/Auto Restart

When this optional feature detects a power failure, it preserves system status, and brings the system to an orderly halt. When power is restored, the system will restart if the power switch on the console is in the LOCK position. If the switch is not in LOCK, the system must be manually restarted at location 0. The system will then close all user files and each user must press ESC to activate his terminal. BASIC will then print:

```
POWER FAIL [AT nnnn]
*
```

at the terminal, where nnnn is the statement which was being executed when the power failure occurred. User programs will remain intact.

Error Messages

The error messages which can be output when operating an SOS BASIC system differ from those listed in Appendix A for RDOS systems.

The SOS BASIC error messages are given in Tables 4-4 and 4-5.

Table 4-4. Error Messages (Initiated by SOS BASIC)

Error Code	Meaning
---	INCOMPATIBLE OPERATING SYSTEM
---	NO CORE
00	ARITHMETIC OPERATORS IN ILLEGAL COMBINATION
01	INVALID CHARACTER
02	SYNTAX
03	[MAT] READ/DATA TYPES INCONSISTENT
04	INTERNAL SYSTEM FAULT
05	INVALID STATEMENT NUMBER
06	ATTEMPT TO DEFINE MORE THAN 93 VARIABLES
07	ILLEGAL COMMAND (FROM A FILE)
08	PAGE OR TAB SPECIFICATION ILLEGAL
09	ILLEGAL RESERVED FILENAME
10	RESERVED FILE IN USE
11	PARENTHESIS NOT PAIRED
12	ILLEGAL COMMAND
13	STATEMENT NUMBER MISSING
14	INSUFFICIENT STORAGE TO ENTER STATEMENT
15	UNSATISFIED [MAT] READ
16	ARITHMETIC OVERFLOW, UNDERFLOW OR DIVIDE BY ZERO
17	UNDEFINED VARIABLE
18	GOSUB NESTING LIMIT
19	RETURN - NO GOSUB
20	FOR NESTING LIMIT
21	FOR - NO NEXT
22	NEXT - NO FOR

Error Code	Meaning
23	INSUFFICIENT STORAGE FOR A VARIABLE OR AN ARRAY
24	LINE NUMBER MISSING
25	MAT OR PRU NOT IN SYSTEM
26	INSUFFICIENT STORAGE TO LOAD SAVE FILE
27	INVALID FILE REFERENCE
28	ARRAY EXCEEDS INITIAL DIMENSION
29	EXPRESSION TOO COMPLEX FOR EVALUATION
30	INVALID FILE MODE
31	SUBSCRIPT EXCEEDS DIMENSION
32	UNDEFINED USER FUNCTION
33	FUNCTION NESTING LIMIT
34	FUNCTION ARGUMENT
35	ILLEGAL EDIT MASK
36	PRINT LINE GREATER THAN PAGE WIDTH
37	USER SUBROUTINE (SBRTB) NOT FOUND
38	UNDIMENSIONED STRING
39	REDUNDANT MATRIX SPECIFICATION
40	MATRICES UNEQUAL SIZES
41	MATRIX HAS ONLY ONE DIMENSION
42	FILE ALREADY OPENED
43	MATRIX NOT SQUARE
44	FILE NOT OPEN
45	NOT A SAVE FILE
46	INCORRECT RESPONSE TO [MAT] INPUT
47	FILE OPENED IN WRONG MODE

Table 4-5. SOS Error Messages (Initiated by SOS)

Error Code	Message
00	ILLEGAL CHANNEL
01	ILLEGAL FILENAME
02	ILLEGAL SYSTEM COMMAND
03	ILLEGAL COMMAND FOR DEVICE
06	END OF FILE
07	READ-PROTECTED FILE
08	WRITE-PROTECTED FILE
10	NONEXISTENT FILE
13	FILE NOT OPEN
17	CHANNEL IN USE
18	RECORD SIZE EXCEEDED
20	PARITY
22	STORAGE ALLOCATION
24	FILE DATA CHECK
25	UNIT IMPROPERLY SELECTED
30	ILLEGAL DEVICE CODE
33	ILLEGAL TIME OR DATE
37	INTERRUPT DEVICE CODE IN USE

Differences Between RDOS BASIC and SOS BASIC

SOS Extended Basic:

- has no MMPU support
- has no floating point unit support
- has no string arithmetic
- has no VAL or STR\$ functions
- has no disk support (swapping, accounting, BATCH, disk files, etc.)
- requires that the system console be the same as the SOS CLI console
- has no account names
- refers to system console line number as 'SC' in sign-on and sign-off messages
- has no LIBRARY, FILES or WHATS commands
- has manual restart capability
- does not print a prompt at each terminal on start-up
- has a time slice of 16/100 seconds
- runs the real-time clock at 100 Hz
- supports a 4026 Asynchronous Line Multiplexor
- has only 96 variables
- starts arrays at zero
- truncates strings on short substring assignments

End of Chapter

Appendix A Error Messages

The system prints Extended BASIC error messages as two-digit codes, followed by a brief explanatory message. There are three categories of errors which may occur when operating BASIC.

1. *Errors recognized by BASIC during program input (Table A-1).*

If an error is detected in a statement input from a terminal, the error message refers to the last statement typed.

If the statement in error was input from a file or other input device, BASIC prints the incorrect statement followed by the error message.

All syntax errors are recognized during program input.

The form of the error message is:

ERROR xx text

xx: a two-digit decimal error code.

text: a brief description of the error.

2. *Run-time errors (except file I/O) (Table A-1).*

BASIC system run-time errors cause printout of an error message in the following form:

ERROR [xx AT yyy] text

xx: a two-digit decimal error code.

yyy: the line number at which the error occurred, if used in a statement.

text: a brief description of the error.

3. *Extended BASIC File I/O errors (Table A-2).*

The format for file I/O error messages is as follows:

I/O ERROR xx [AT yyy] text

xx: a two-digit decimal error code.

yyy: the line number at which the file I/O error occurred, if used in a statement.

text: a brief description of the error.

4. *Initialization Errors (Table A-3)*

Some errors occur only during the initialization process that follows the command to invoke BASIC. The form of these error messages is:

ERROR xx text

xx: a two-digit decimal error code.

text: a brief description of the error.

Table A-1. BASIC Error Messages

Code	Error Message	Meaning
00	FORMAT	Unrecognizable statement format.
01	CHARACTER	Illegal ASCII character or unexpected character.
02	SYNTAX	Invalid argument type.
03	READ/DATA TYPES	READ specifies different data type than DATA statement.
04	SYSTEM	Hardware or software malfunction.
05	LINE NUMBER	Statement number not in the range $1 \leq n \leq 9999$.
06	EXCESSIVE VARIABLES	Attempt to declare too many variables.
07	COMMAND	Attempt to execute an illegal command.
08	SINGULAR MATRIX	Attempt to invert a singular matrix.
09	(not used)	
10	ATTRIBUTE	Attempt to assign an illegal attribute to a file.
11	PARENTHESIS	Parentheses in an expression are not paired.
12	(not used)	
13	(not used)	
14	PGM OVFL	Not enough storage to ENTER source program.
15	END OF DATA	Not enough DATA arguments to satisfy READ.
16	ARITHMETIC	Value too large or too small to evaluate; or a divide by 0.
17	(not used)	
18	GOSUB NESTING	More nested GOSUBs than specified at BASIC system generation.
19	RETURN - NO GOSUB	RETURN statement encountered without a corresponding GOSUB.
20	FOR NESTING	More nested FORs than specified at BASIC system generation.
21	FOR - NO NEXT	Unexecutable FOR-NEXT loop; FOR without a NEXT.
22	NEXT - NO FOR	NEXT statement encountered without a corresponding FOR.
23	DATA OVFL	Not enough storage left to assign space for variables.
24	DIRECTORY EMPTY	No files in your directory.
25	OPTION	Feature specified not available.
26	(not used)	
27	FILE NUMBER	Invalid file designation in an I/O statement.
28	DIM OVFL	An array or string exceeds its original dimensions.

Table A-1. BASIC Error Messages (continued)

Code	Error Message	Meaning
29	EXPRESSION	An expression is too complex for evaluation.
30	ILLEGAL FILE MODE	Invalid mode designation in an I/O statement.
31	SUBSCRIPT	Subscript exceeds array's dimensions.
32	UNDEFINED FUNCTION	Attempt to use a function never defined by DEF.
33	FUNCTION NESTING	User function nesting exceeds BASIC systems generation specification.
34	FUNCTION ARGUMENT	Argument range out of bounds.
35	ILLEGAL MASK	PRINT USING format is illegal.
	(not used)	
37	USER ROUTINE	CALL statement specifies a user routine not in storage.
38	(not used)	
39	DUP MATRIX	Same matrix appears on both sides of a MAT transpose statement.
40	MATRICES SIZES	Matrices have different sizes.
41	UNDIMENSIONED VARIABLE	Attempt to use an undimensioned matrix.
	(not used)	
43	MATRIX NOT SQUARE	Attempt to invert a nonsquare matrix.
	(not used)	
45	DATA > LRECL	Logical record length limit exceeded.
46	INPUT	Too many responses to [MAT]INPUT.
	(not used)	
	(not used)	
49	NO ROOM FOR DIRECTORY	A FILE or LIBRARY command cannot find 256 words in your program storage area to read the disk directory.
	(not used)	
51	USER NOT ACTIVE	Attempt to send message to an inactive or nonexistent user.
52	USER IN NOMSG STATE	Attempt to send message to user whose terminal is in NOMSG state.
	(not used)	
54	STATEMENT LENGTH	A statement exceeded 132 characters in either internal or ASCII format, when expanded.
55	EXECUTE-ONLY	Attempt to examine a program originating from a file with the execute-only attribute.

Table A-1. BASIC Error Messages (continued)

Code	Error Message	Meaning
56	RANGE	Attempt to reference a random record beyond 262,144.
57	(not used)	
58	INCOMPATIBLE CORE IMAGE	Attempt to LOAD a core image file SAVED under a different floating-point precision.
59	ZERO STEP	FOR-NEXT with step 0.
60	TIME-OUT	Timed input decremented to 0.
61	INVALID DECIMAL STRING	Attempt to perform string arithmetic with nonnumeric characters.
62	STAR OVFL	The result of string arithmetic requires more than 18 digits for precision representation.
63	(not used)	
64	SYSTEM ACTIVE	Attempt by system manager to execute a BYE command while users were still on system.

Table A-2. Extended BASIC File I/O Error Messages

Code	Error Message	Meaning
01	ILLEGAL FILE NAME	A to Z, 0 to 9 and \$ are only valid characters.
02	ILLEGAL SYSTEM COMMAND	Command not defined in operating system.
03	ILLEGAL COMMAND FOR DEVICE	INIT "\$PTR", WRITE to \$CDR.
04	NOT A CORE IMAGE FILE	File not in SAVE format.
06	END OF FILE	Attempt to read beyond EOF marker.
07	READ PROTECTED FILE	Attempt to read from a read-protected file.
08	WRITE PROTECTED FILE	Attempt to write a write-protected file.
09	FILE ALREADY EXISTS	Attempt to create an existent file.
10	FILE NOT FOUND	Attempt to reference a nonexistent file.
11	PERMANENT FILE	Attempt to alter a permanent file.
12	ATTRIBUTE PROTECTED	Attempt to change file attributes when file is protected with RDOS attribute A.
13	FILE NOT OPENED	Attempt to reference an unopened file.
14	SWAPPING DISK DATA CHECK	Disk error on swapping file.

Table A-2. Extended BASIC File I/O Error Messages (continued)

Code	Error Message	Meaning
15	REVISION CHECK	Core image file is not compatible with this system.
16	CHECKSUM	Disk error.
17	CHANNEL NOT AVAILABLE	Attempt to open too many files. Operating system file pool overflowed.
18	LINE LIMIT	Line limit exceeded on read or write line.
20	PARITY	Parity error on read line.
23	NO FILE SPACE	Out of disk space. Delete files to make more room.
24	READ ERROR	File read error.
25	SELECT STATUS	Unit not ready or is write-protected.
29	DIFFERENT DIRECTORIES	Files specified on different directories.
30	ILLEGAL DEVICE CODE	Device not in system or illegal device code.
31	ILLEGAL OVERLAY	This is an unexpected system software error. If it occurs, please notify your local Data General representative.
37	DEVICE ALREADY INITIALIZED	Device already INITed.
38	INSUFFICIENT CONTIGUOUS BLOCKS	Insufficient number of free contiguous disk blocks. Reorganize partition.
41	NO MORE DCB'S	Attempt to open more devices or directories than are configured in the operating system.
42	ILLEGAL DIRECTORY SPECIFIER	Illegal directory specifier.
43	UNKNOWN DIRECTORY SPECIFIER	Directory specifier unknown.
44	DIRECTORY TOO SMALL	Directory is too small (Operator only). Minimum directory size is 48 blocks (CPART command).
45	DIRECTORY DEPTH	Directory depth exceeded (Operator only).
46	DIRECTORY IN USE	Attempt to release a directory in use by other program.
47	LINK DEPTH	Link depth exceeded.
48	FILE IN USE	Use RDOS CLEAR command.
52	FILE POSITION	Attempt to read out of bounds.
54	DIRECTORY NOT INITIALIZED	Directory/device not initialized by INIT command.
58	DIRECTORY SHARED	No file space left.
69	DISK IS FULL	No file space left.

Table A-3. Initialization Errors

Error Message	Explanation
ERROR 01-REAL-TIME CLOCK	Real-time clock not in operating system or generated at other than 10 Hz.
ERROR 02-OVERLAY FILE	The overlay file (filename.OL) for the invoked BASIC system cannot be opened. It may have been deleted, altered or renamed. In any case, it is necessary to generate another BASIC system as described in Chapter 2. This error may also be caused by a disk data error which would require rebuilding the RDOS system and that BASIC system files be restored as described in Chapter 2.
ERROR 03-INSUFFICIENT CHANNELS	This is an unexpected system software error. If it occurs, please notify your local Data General representative.
ERROR 04-INSUFFICIENT FREE STORAGE	name.CL may have been altered.
ERROR 05-SWAPPING FILE	A BASIC system, configured for swapping, cannot open file BASIC.SW. Try the RDOS CLEAR command to reset the file's use count to 0. If that doesn't work it may be necessary to generate a BASIC system which allocates each user less memory space. Another alternative is to DUMP the files on the swapping device to magnetic tape for back-up and then selectively DELETE files from the swapping device to create more file space.
ERROR 06-INSUFFICIENT TCB'S	This is an unexpected system software error. It can be caused if the BASIC.CL file, created by the BASIC system generation program, is altered before it is used to generate the system (i.e., @BASIC.CL@). If this is not the cause, contact your local Data General representative.
ERROR 07-MULTIPLEXOR	BASIC uses the RDOS multiplexor handlers for supporting the 4060 and ALM multiplexors. Therefore, the current RDOS system was generated without QTY or ALM device in the system. Generate a new RDOS system which includes QTY or ALM support.
ERROR 08-MASTER DIRECTORY	BASIC.DR is not in system. BASIC requires a directory named BASIC.DR for use as a public directory. This directory contains the files whose names are listed when the BASIC command LIBRARY is given. Use the RDOS CDIR or CPART commands to create a subdirectory or secondary partition in the master device primary partition.
ERROR 09-ID FILE	No BASIC.ID file in system or BASIC.ID in use.
ERROR 10-ACCOUNTING FILE	BASIC.AF file in use.
ERROR 12-EXTENDED MEMORY	This is an unexpected system software error. If it occurs, please notify your local Data General representative.
<p>In addition to the BASIC initialization errors listed above, the following RDOS system error can occur during BASIC initialization.</p>	
Error Message	Explanation
NO ROOM FOR UFT'S:filename	The number of channels requested by BASIC is more than the number available from the operating system. Either generate a new operating system with more channels, or generate a new BASIC which requires fewer channels. BSG query 14 (Chapter 2,) explains the method for calculating system channel requirements for a BASIC system.

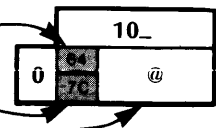
End of Appendix

Appendix B ASCII Character Set

To find the *octal* value of a character, locate the character, and combine the first two digits at the top of the character's column with the third digit in the far left column.

LEGEND:

Character code in decimal
EBCDIC equivalent hexadecimal code
Character



OCTAL	00_	01_	02_	03_	04_	05_	06_	07_
0	NUL	BS (BACK-SPACE)	P	X	SPACE	(0	8
1	A	HT (TAB)	Q	Y	!)	1	9
2	B	LINE FEED	R	Z	" (QUOTE)	*	2	:
3	C	VT (VERT. TAB)	S	ESC (ESCAPE)	#	+	3	;
4	D	FF (FORM FEED)	T	\ 	\$, (COMMA)	4	<
5	E	CR (RETURN)	U] 	%	-	5	=
6	F	N	V		&	. (PERIOD)	6	>
7	BELL G	O	W	_	' (APOS)	/	7	?

OCTAL	10_	11_	12_	13_	14_	15_	16_	17_
0	~	H	P	X	` (GRAVE)	h	p	x
1	A	I	Q	Y	a	i	q	y
2	B	J	R	Z	b	j	r	z
3	C	K	S	[c	k	s	}
4	D	L	T	\	d	l	t	
5	E	M	U]	e	m	u	}
6	F	N	V	or	f	n	v	~ (TILDE)
7	G	O	W	- or	g	o	w	DEL (RUBOUT)

SD-00476 Character code in octal at top and left of charts

* means CONTROL

End of Appendix

Appendix C Privileged Command Summary

Formats and Descriptions	S	C
BYE Exits from the BASIC system to the operating system CLI.	√	√
CDIR <i>[device specifier:] name</i> Creates a subdirectory in the current master directory or in any other primary or secondary partition.	√	√
CPART <i>[device specifier:] name, size</i> Creates a secondary partition in the master directory or in any other primary partition.	√	√
DIR { <i>primary part [:secondary part][:subdir]</i> } { <i>secondary part [:subdir]</i> } Changes the current directory.	√	√
DISABLE Inhibits transmission of the CTRL-A and CTRL-C system break characters.	√	√
ENABLE Re-enables transmission of the CTRL-A and CTRL-C system break characters.	√	√
FREE userID Interrupts program execution for userID.	√	√
INIT { svar } { "str lit" } Initialize a directory or a device (MTn or CTn).	√	√

Formats and Descriptions	S	C
KILL userID Forces a user (userID) off the system.	√	√
LEVEL <i>[userID][n]</i> Monitors or sets priority levels.	√	√
MAX = val Establish a limit for the number of active system users.	√	√
RELEASE { svar } { "str lit" } Release a directory or a device (MTn or CTn).	√	√
USERS Prints a list, on the system console, of active system users.	√	√
ALL message Transmit a message to all active users who have not set NOMSG.	√	√
FALL message Forces message transmission to all users including those who have set NOMSG.	√	√
FMSG userID message Forces message transmission to userID even though NOMSG may have been set. NOTE: MSG command same as for users.	√	√

End of Appendix

Appendix D

Sample BSG Dialog

Extended BASIC Revision 05.00 08/15/77 SYSTEM GENERATION

VALID ANSWERS ARE IN PARENTHESES. DEFAULT VALUES ARE ALWAYS GIVEN FIRST AND MAY BE SELECTED BY A NULL RESPONSE (CARRIAGE RETURN). DEFAULTS ARE FOLLOWED BY A LIST OF ALTERNATIVES, ANY ONE OF WHICH MAY BE SELECTED BY TYPING A MATCHING RESPONSE.

OBJECT COMPUTER TYPE (NOVA,ECLIPSE,NOVA3,ORIGINAL) ? NOVA
 OBJECT OPERATING SYSTEM (RDOS,RTUS,DOS) ?
 SYSTEM SAVE FILE (BASIC, ANY DISK FILE NAME) ? JERRY
 ACCOUNTING (YES,NO) ? YES
 MULTI-USER SYSTEM (NO,YES) ? YES
 LINE CONFIGURATION: 0-7
 MASTER CONSOLE LINE NUMBER (-1,0-31) ?
 IS MASTER CONSOLE A CRT (YES,NO) ? YES

CENTRAL PROCESSOR OPTION SUPPORT:
 DIAL-UP LINES (NO,ALM,4M60) ?
 SWAPPING (YES,NO) ? YES
 HARDWARE MULTIPLY/DIVIDE (NO,YES) ? YES
 HARDWARE FLOATING POINT (NO,SING,DOUB) ? DOUB
 MEMORY MANAGEMENT/PROTECTION (NO,YES) ? YES

USER WRITTEN SUBROUTINES (NO,YES) ? YES
 DEFAULT FEATURES ACCEPTABLE (YES,NO) ?
 DEFAULT OVERLAYS ACCEPTABLE (YES,NO) ? NO

VALID ANSWERS TO THE FOLLOWING ARE:

O=INCLUDE FEATURE AS AN OVERLAY
 M=INCLUDE FEATURE IN MAIN STORAGE
 NO=DO NOT INCLUDE FEATURE

MATRIX ARITHMETIC (O,M,NO) ? M
 MATRIX INPUT AND OUTPUT (O,M,NO) ? O
 MATRIX INVERSE (O,M,NO) ?
 PRINT USING (O,M,NO) ? M
 BYE (O,M) ?
 CALL (O,M) ? O
 MARK SENSE CARD READER (O,M,NO) ?
 LIST,PUNCH (O,M) ?
 FILE COMMANDS (O,M,NO) ?
 USERS COMMAND (O,M) ?
 OPERATOR COMMANDS (O,M) ?
 NEW (O,M) ?
 USER INITIALIZATION (O,M) ?
 DELETE,RENAME,PAGE,RENUMBER,SIZE,INIT,RELEASE (O,M,NO) ?
 CONSOLE CHARACTERISTICS (O,M,NO) ?
 DISK AND DIRECTORY COMMANDS (O,M,NO) ?
 STRING ARITHMETIC (O,M,NO) ? M
 ERROR MESSAGE TEXT (O,M,NO) ?
 I/O ERROR MESSAGE TEXT (O,M,NO) ?
 SIN,COS,TAN,ATN (O,M,NO) ?
 LOG,EXP,F,SGR,RND (O,M,NO) ?
 LEN,POS,STR\$,VAL (M,NO) ?
 RUN FILENAME,CHAIN,SAVE,LOAD,LOGON (O,M,NO) ?
 MAXIMUM FILES TO BE OPEN AT ONE TIME (10,1-81) ?
 FILE SHARING (YES,NO) ?
 FOR-NEXT NESTING LIMIT (8,4-20) ? 5
 GOSUB NESTING LIMIT (8,4-20) ? 14
 USER FUNCTION NESTING LIMIT (6,0-10) ? 2
 MAXIMUM USER CORE SIZE (IN BYTES) ? 5000
 PROMPT STRING (*-SPACE=NULL, ANY 10 CHARACTER STRING) ? READY
 TIME OF DAY WITH PROMPT (NO,YES) ? YES

End of Appendix

Index

Italicized page numbers indicate primary references to commands. The legend “f” after a page number means “and the following page”; “ff” means “and the following pages”.

- abbreviations in formats iv
- access controlling 3-15
- accounting information 2-7f
- ALL command 3-5
- ASCII Character Set B-1
- attributes, file 3-16
- auto restart procedures 3-4; (SOS 4-7)

- backup procedures 3-16
- BASIC errors
 - see errors
- BASIC files,
 - see file, names
- BASIC, loading 2-5
- BASIC-supported devices 1-1
- BASIC System Generation,
 - see BSG
- BASIC.AF file 2-7f, 3-4
- BASIC.CL file 2-5
- BASIC.DR directory 2-6
- BASIC.ID file 2-6, 3-15
- BASIC.OL file 2-2
- BASIC.SV file 2-2
- BSG 2-1, Chapter 2
- BSG Dialog 2-1 to 2-4, Appendix D
- BATCH 3-16f
- breaks, RDOS console 3-3, 3-7f
- BYE command 3-3, 3-5

- CDIR command 2-6, 3-6
- changing directories 3-3
- CHATR command 3-15
- CLI prompt (R) 3-2
- CLI/RDOS/BASIC interface concepts 1-1f, 2-1f, 2-4, 3-1, 3-15ff

- cold start 3-1f
- command format iii, iv
- Command Summary
 - Appendix C
- commands or statements
 - see Table of Contents, and Index
- Command Line Interpreter
 - see CLI
- communicating with users 3-3
- CPART command 2-6, 3-6
- crash, system 3-4
- CTRL-A 3-3, 3-7f
- CTRL-C 3-3, 3-7f
- CTRL-F 3-3, 3-7f

- device filenames 4-6
- device independence 3-2
- device, master 2-5f
- DIR command 2-6, 3-7, 3-15
- directory 2-5f
- directory, BASIC.DR 2-6f
- directory, current 2-5, 3-3, 3-7, 3-14
- directory, master 3-6
- directory sharing 3-15
- DISABLE command 3-3, 3-7
- disk partitions and directories 2-5ff
- documentation conventions iii
- documents, reference list iii
- DUMP command 3-16

- ENABLE command 3-3, 3-8
- ENTER command 3-17
- error, initialization 3-2
- error messages, RDOS BASIC
 - during input A-1 to A-4
 - file I/O A-4ff
 - initialization A-6
 - runtime A-1 to A-4
- error messages, SOS BASIC 4-7ff
- ESC command 2-6
- extended memory support 1-2

- failure, system 3-4
- FALL command 3-8, 3-15
- File, names:
 - BASIC.AF 2-7f
 - BASIC.CL 2-2
 - BASIC.DR 2-6
 - BASIC.ID 2-6f, 3-15
 - BASIC.OL 2-2
 - BASIC.SV 2-2
 - HELLO.SV 2-7
 - library 2-6
 - overlay 2-2
 - BASIC system save 2-1, 3-1, 3-16
- file space, disk 2-5f
- FMSG command 3-9, 3-15
- foreground/background 1-2, 3-1, 3-3, 3-7
- FOR-NEXT nesting option 2-4
- FREE command 3-9

- GDIR command 3-10
- GOSUB nesting option 2-4

- hardware, minimum configuration 1-1
- HELLO.SV file 2-7

- identification code 3-15
- identification file 2-6f, 3-15
- initialization by BASIC
 - command 3-10
 - errors A-6
- INIT command 2-6, 3-10
- invoking BASIC 3-2

- KILL command 3-11

- LEVEL command 3-11
- library, BASIC system 2-6
- loading BASIC under RDOS 2-5
- log-on procedure 2-7
- log-on program(s) 2-7
 - HELLO.SV 2-7

- manuals, user and reference iii
- mark-sense cards 2-3, 3-16f
- matrix, BSG options 2-3
- matrix manipulation key words
 - see Appendix C
- MAX command 3-12
- memory, maximum, minimum 1-2
- modules, loading 2-1
- monitoring the system 3-15
- MSG command 3-3, 3-12
- multiplexors (ALM and QTY) 2-2, 3-1
- multiuser system 1-1, 2-2, 2-6ff, 3-16

- operator responsibilities 3-1
- options during BSG 2-1 to 2-4
- overlays 2-2

- partition 2-5
- partition, primary 2-5
- partition, secondary,
 - see Secondary partition
- password 3-16
- power fail procedures 3-4
- privileged commands 3-4 to 3-14, Appendix C
- program protection 3-15

- recovery from system consoles 3-4
- RELEASE command 3-13
- restart procedure 3-4; SOS 4-6f
- RDOS errors
 - see errors
- RDOS/BASIC interface
 - see CLI
- RDOS/MRDOS 1-1f
- relocatable load
 - for SOS BASIC 4-3ff
 - for RDOS BASIC 4-5

- secondary partition 2-5f, 3-10
- secondary partition, creating (CPART) 3-6
- security system 3-15
- sign-on (SOS) 4-6
- Sort/Merge 1-1
- SOS BASIC Chapter 4
- SOS BASIC paper tapes 4-2f
- SOS driver symbols 4-2
- SOS error messages
 - see errors
- SOS SYSGEN Chapter 4
- statement, format iii
- statements
 - see Table of Contents and Index
- stopping the BASIC System 3-3
- subdirectory 2-5f
- subdirectory, creating (CDIR) 3-6
- swapping file 2-2
- symbols, documentation iiif
- SYSGEN dialog, BSG (RDOS) 2-2 to 2-4
- SYSGEN, RDOS 2-1
- SYSGEN, SOS Chapter 4
 - dialog 4-5f
- SYSIN 3-16f
- SYSOUT 3-16f
- system console (SC) 3-1
- system console breaks (RDOS) 3-3
- system crashes 3-4
- system hardware 1-1

system initialization
 see Initialization
system, multiuser 1-1, 2-2, 2-6ff, 3-16
system software 1-1
system, timesharing
 see timesharing system

/T Switch 3-17
timesharing system 1-1, 2-4 to 2-7, 3-16f
terminal, definition of iv
Trigger (SOS) 4-1f

USERS command 3-13f
user directories 2-5ff

WHATS command 3-16

How Do You Like This Manual?

Title _____ No. _____

We wrote the book for you, and naturally we had to make certain assumptions about who you are and how you would use it. Your comments will help us correct our assumptions and improve our manuals. Please take a few minutes to respond.

If you have any comments on the software itself, please contact your Data General representative. If you wish to order manuals, consult the Publications Catalog (012-330).

Who Are You?

- EDP Manager
- Senior System Analyst
- Analyst/Programmer
- Operator
- Other _____

What programming language(s) do you use? _____

How Do You Use This Manual?

(List in order: 1 = Primary use)

- _____ Introduction to the product
- _____ Reference
- _____ Tutorial Text
- _____ Operating Guide
- _____ _____

Do You Like The Manual?

Yes	Somewhat	No	
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Is the manual easy to read?
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Is it easy to understand?
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Is the topic order easy to follow?
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Is the technical information accurate?
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Can you easily find what you want?
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Do the illustrations help you?
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Does the manual tell you everything you need to know?

Comments?

(Please note page number and paragraph where applicable.)

From:

Name _____ Title _____ Company _____
 Address _____ Date _____

FOLD DOWN

FIRST

FOLD DOWN

FIRST
CLASS
PERMIT
No. 26
Southboro
Mass. 01772

BUSINESS REPLY MAIL

No Postage Necessary if Mailed in the United States

Postage will be paid by:

Data General Corporation

Southboro, Massachusetts 01772

ATTENTION: Software Documentation

FOLD UP

SECOND

FOLD UP