SYSGEN

DOS/65 SYSTEM GENERATOR

VERSION 2.1

TABLE OF CONTENTS

SECTION 1 - INTRODUCTION

The program SYSGEN.COM provides the DOS/65 user with the ability to create a "new" version of DOS/65 which includes the following:

- User specified memory size
- User specified SIM length
- User created BOOT
- User created SIM
- User specified start of available page zero RAM

NOTE

The term "memory size" refers to the length of memory as measured from $0000. If for example the last RAM location in a system is $9FFF the "memory size" is $9FFF+1 or 40K (K=1024). This value is independent of the system. Thus a KIM-1 with 32K of RAM starting at $2000 is a 40K system just as an Apple II with RAM running continuously from $0000 is also a 40K system. In all cases the region from TEA start to "memory size"-1 must be RAM.

SECTION 2 - OPERATIONS

2.1 PREPARATION

Prior to execution of SYSGEN the user must prepare separate BOOT.ASM and SIM.ASM files using the DOS/65 editor EDIT.COM or other suitable text editor and must then assemble these two routines to prepare the appropriate BOOT.KIM and SIM.KIM files. The files do not have to be named "BOOT" and "SIM" since SYSGEN will ask for the name portion of the file designation when it is needed. These object code files must, however, be of type "KIM" or they will not be read by SYSGEN.

While examples of these programs are provided as part of the DOS/65 documentation and the key functional requirements of those programs are discussed in the SYSTEM INTERFACE GUIDE a few summary points are listed in Table 2-1 to ensure that the critical requirements are checked before using SYSGEN. SYSGEN does not (and in most cases cannot) check that the routines meet all these requirements. The one key item that is checked is the origin and length of the object code ("KIM") file. If errors of that nature are detected during the read of either file, the run will be aborted and an error message displayed on the console.

Within the SYSGEN (v2.16 and later) source code there is a provision made to allow the user to specify the start of available page zero RAM for SYSGEN. The assembly time parameter PZSTRT must be set to whatever value is correct for the user's system. The SYSTEM INTERFACE GUIDE (section B.2) discusses this parameter in depth. Normally this is set to $2 but if it must be changed then SYSGEN must be assembled and a binary created with the correct page zero memory allocation in accordance with PZSTRT. This may require cross-platform development or help from the developer.

2.2 EXECUTION

The program is executed by entering a CCM command line of the general form:

```
      (drive:)SYSGEN
```

The optional (drive:) term is only needed if SYSGEN is to be loaded from a drive other than the default drive.

2.2.1 STEP 1 - MEMORY SIZE (MSIZE)

The first entry required by the user is specification of the memory size in K (K = 1024 bytes) for which the new system is to be configured. This number must be a decimal integer and must be less than or equal to 64 (K). The lower limit is a function of the TEA start address as shown in Table 2-2. These limits are set so that the longest "normal" programs will fit into the TEA without conflict with CCM, PEM, or SIM. While it might be possible in some circumstances to use a smaller minimum, SYSGEN will not allow that to happen.

The main reason for that restriction is that one of the longer programs is SYSGEN itself and if the new system is too small the next attempt to use SYSGEN might not work. The limitations are a form of self-protection to ensure that the user will be able to continue to create new systems.

TABLE 2-1 Key BOOT & SIM Features

	BOOT	**SIM**
ORIGIN	$0000	(MSIZE*1024)-(PAGES*256) where MSIZE is memory size in K (K=1024) and PAGES is SIM length in pages (page=256 bytes)
LENGTH	<= 128 bytes	256 or more bytes in 256 byte increments
ENTRY	Determined by user loader but usually $0000	Multiple entries organized as 18 JMP's and one data block beginning at the origin. See SYSTEM INTERFACE GUIDE for details.
EXIT	Jump to SIM after loading CCM, PEM and SIM.	Returns to caller by executing RTS after executing desired function. If error reset system. (Exceptions are actions after entering at cold and warm boot vectors at SIM and SIM+3. See SYSTEM INTERFACE GUIDE.)
STATUS	Volatile. Can be destroyed as soon as function completed.	Must not be destroyed. Is the only way DOS/65 has of talking to the user's hardware & software.
PARAMETERS	None	Usually is a single byte passed in A but is sometimes a 16 bit value in A (low) and Y (high).
RETURN	None	Usually is a single byte in A but is sometimes a 16 bit value in A (low) and Y (high). One function uses X for part of the return value.

Table 2-2 Minimum Memory Size	
TEA Start	**Minimum Memory Size**
$200	16K
$400	16K
$800	16K
$1000	20K
$1400	21K
$2000	24K

2.2.2 STEP 2 - SIM LENGTH (PAGES)

The program will next ask the user to enter the length of SIM in pages (page = 256 bytes) for which the new system is to be configured. This entry must be a decimal integer between 1 and 99 inclusive. This number is used in conjunction with the Memory Size (MSIZE) to calculate the load addresses of CCM, PEM and SIM as shown in the following 6502 assembly language:

```
Simlng  =       pages*256
Sim     =       msize*1024-simlng
Pem     =       sim-3072
Ccm     =       pem-2048
```

Similar equates must be used in BOOT and SIM to properly place or address the DOS/65 modules. SYSGEN does check the memory organization of the new system to make sure that SYSGEN will run under that configuration. If the combination of MSIZE and PAGES is such that a SYSGEN of the same configuration could not be performed, a memory size error message will be printed and execution will be terminated.

2.2.3 STEP 3 - NEW BOOT

The program will next ask the user to enter the drive and name field of the UFN for the "BOOT.KIM" file. Typical responses to this prompt include:

RESPONSE	MEANING
`TRBLBOOT`	`Use file TRBLBOOT.KIM on default drive`
`b:bootnew`	`Use file BOOTNEW.KIM on drive B`

The user should ensure that the proper diskette is inserted into the desired drive before completing this response as the program immediately begins the load. If SYSGEN cannot find the file, the run will be aborted.

During the read of the file, SYSGEN checks to ensure that only addresses in the proper range exist in the "KIM" file. If improper addresses (see Table 2-2) are found or if a read error occurs, the run will also be aborted. If that happens it is suggested that the user check the "KIM" file (it is a normal ASCII text file) and make sure the data is good and that the addresses are valid.

2.2.4 STEP 4 - NEW SIM

The flow in this step is identical to that of Step 3. The key difference from the user's point of view is that the job of making sure that the "KIM" file addresses are correct is a bit more difficult. Unlike BOOT, the location of SIM is a function of memory size and SIM length and thus its origin must be changed as a function of those parameters. The example shown in Section 2.2.2 shows one way to do this. As was the case for BOOT, SYSGEN does check that the correct addresses are used and will abort the run if an error is detected.

The SIM file input to SYSGEN can be any length as long as it does not overwrite CCM and the resulting data will fit on the system tracks. SYSGEN will write the full number of system tracks allocated by the DCB of the designated drive. If the SIM file included data that would go beyond the allocated system tracks, execution will be aborted.

The user can use these features to allocate data areas at the top of SIM. If these areas do not contain code, the user does not have to allocate system track space for their use. The following example illustrates this capability.

```
                            SIM = 7 PAGES

                            a        b
                            .        .
                            .        .
                            v        v

              ..... ..... ..... ..... ..... ..... .....
              .               . .                     .
              ................. .......................
                      .                    .
                      .                    .
                      .                    .

3 PAGES OF DATA FROM KIM FILE       4 PAGES OF
                                    UNINITIALIZED DATA AREA
```

If the space available on the system tracks only allows the data through "a" to be written to disk, SYSGEN will issue an error message and terminate execution. If, however, the system tracks could hold the data through "b" then SYSGEN will execute normally. In either case the user's code in BOOT and SIM must keep track of the load addresses and the number of tracks/sectors to load.

2.2.5 STEP 5 - DESTINATION DRIVE

In the final step the user is asked to specify the drive on which the new system will be written. While SYSGEN will recognize as correct responses "A" to "H" or "a" to "h", the user should ensure that the specified drive is actually in place. In addition, it is essential that the DCB for the specified drive be in place and that the DCB contents be compatible with the new system configuration. SYSGEN uses this data to determine the number of system tracks available and the number of sectors in each track.

As before, the user must also ensure that the correct diskette is in the desired drive before completing this response.

After writing the new system, SYSGEN will return to CCM by executing a RTS. Because neither a WARM nor COLD BOOT is executed, the new system is not in memory and is not being used after execution of SYSGEN. If the new system is to be used, the user should initiate a COLD BOOT using LOADER. (A WARM BOOT does not accomplish the same thing since it will not read in the new SIM!)