

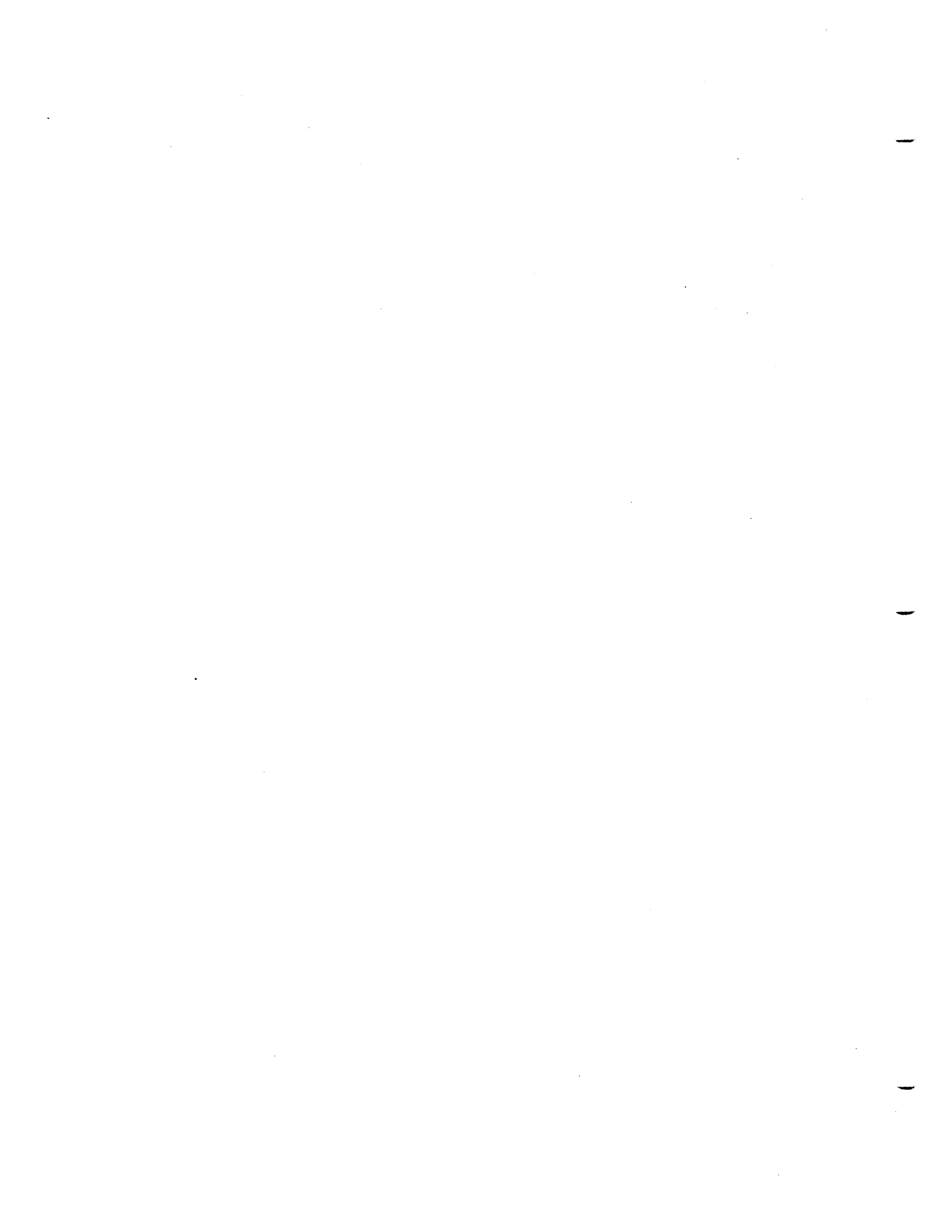
O-FLEX

FLEX™ Operating System for
OS-9 Level II

User's Manual

GIMIX Inc.
1337 W. 37th Place
Chicago, IL 60609
312-927-5510 * TWX 910-221-4055

GIMIX™ and GMX™ are trademarks of GIMIX, Inc. Chicago, IL 60609



GIMIX O-FLEX

CONTENTS

INTRODUCTION	1
SECTION 1: INSTALLING O-FLEX	2
1-1: System Requirements	2
1-2: Installation	2
1-3: Nomenclature	2
SECTION 2: RUNNING O-FLEX	3
2-1: Command Syntax	3
2-2: Returning to OS-9	4
2-3: Console I/O	4
2-4: Floppy Disk Access	4
2-5: File Access	5
2-6: Record Locking	5
2-7: Printer Access	5
2-8: Memory Access	6
2-9: Interrupts	6
SECTION 3: O-FLEX FILES and UTILITIES	7
3-1: FLEX.SYS	7
3-2: XBASPAT.TXT (.BIN)	7
3-3: ASMBPATO.TXT (.BIN)	7
3-4: OFLEX.DEF	7
3-5: OFORM.CMD	8
3-6: SHELL.CMD	8
3-7: ASNDISK.CMD	8
3-8: XCOPY.CMD	9
3-9: TIME.CMD	9
SECTION 4: SOFTWARE COMPATIBILITY	
4-1: Incompatible Utilities	10
4-2: FLEX Diagnostics Package	10
4-3: FLEX Extended BASIC	10
4-4: FLEX Debug Package	10
4-5: Compatibility with Other Software	10
4-6: Patching Software to Run with O-FLEX ...	11

©1984 Frank Hogg Laboratory, Inc
All Rights Reserved
The Regency Tower, Suite 215
770 James St.
Syracuse, NY 13203
(315) 474-7856

This manual is copyright 1983 by FHL, Inc. Reproduction of this manual by any means, in whole or part, without express written permission from FHL, Inc. is prohibited.

GIMIX version ©1984 By GIMIX, Inc.
1337 W. 37th Place
Chicago, IL 60609
(312) 927-5510

FLEX is a trademark of Technical Systems Consultants, Inc.
OS-9 is a trademark of Microware Systems Corp.
GIMIX and GMX are trademarks of GIMIX, Inc.

INTRODUCTION

GIMIX O-FLEX, is a special version of the Technical Systems Consultants FLEX™ Disk Operating System that will run as a user process under the OS-9 GMX II and GMX III versions of Microware's OS-9 Level II. Although O-FLEX is primarily intended for hard disk systems, it can also be used on floppy disk only systems.

O-FLEX creates and uses OS-9 disk files which are structured as complete FLEX disks, and can also access standard GIMIX FLEX format floppy disks. In addition, O-FLEX includes a utility program that allows the exchange of text and binary files between standard FLEX and OS-9 format disks and files. Terminal, and printer I/O is supported by calls to OS-9. All standard FLEX functions, with the exception of printer spooling, are supported by O-FLEX.

O-FLEX is supplied on a standard OS-9 format disk, and includes two major OS-9 files. The file "OFlex", an executable OS-9 module, is the driver package necessary to interface FLEX to OS-9 and a bootstrap loader for FLEX. "FlexSys" is a data file, formatted internally as a FLEX system disk, and contains the actual FLEX, FLEX utility programs, and patch files for some other FLEX programs. FlexSys is equivalent to the system disk normally supplied with stand alone versions of FLEX.

This manual contains information on the installation and use of O-FLEX and the special O-FLEX utilities provided. Included with this manual are copies of the FLEX USER'S and PROGRAMMER'S manuals which contain information on FLEX in general and the standard utilities supplied with O-FLEX.

***** IMPORTANT NOTICE *****

While O-FLEX has been tested by GIMIX and FHL, Inc., and found to function properly with a variety of commonly available software, GIMIX and FHL, Inc. make no guarantees as to its suitability for use in a particular application. Although every effort has been made to make O-FLEX compatible with other versions of FLEX, restrictions exist which may prevent certain software from running under O-FLEX. In particular, software that makes use of, or internally masks interrupts, and software that accesses hardware (other than RAM memory) directly may not function as expected. See SECTION 4 for more information.

O-FLEX is produced by GIMIX, Inc. under license from Technical Systems Consultants. Any questions pertaining to the use or operation of O-FLEX and its associated programs should be directed to GIMIX, Inc. 1337 W. 37th Place, Chicago, Il 60609 (312) 927-5510.

SECTION 1: INSTALLING O-FLEX

1-1: System Requirements

O-FLEX requires a system with either the OS-9 GMX II or OS-9 GMX III operating system and at least 64K of free memory. In addition, Edition 10 or later of the GIMIX floppy disk driver module "G68" MUST be used if access to FLEX format floppy disks is required.

No changes to the configuration of existing systems, other than the installation of the necessary software, are required to run O-FLEX.

1-2: Installation

The following assumes a hard disk based system; for floppy disks, replace "HO" with "DO" as appropriate. O-FLEX assumes a disk system on which a user's working directory is on the same device as the system CMDS directory. Referring to this device as /HO, install the file OFlex as /HO/CMDS/OFlex and install the file FlexSys as /HO/SYS/FlexSys.

To allow multi-user sharing of FlexSys, it must be made sharable with public read permission (attr /HO/SYS/FlexSys pr -s). Since this effectively write-protects the file (see "Record-Locking Considerations" in Section II), any additional files should be installed on FlexSys before setting the attributes as shown above. This also implies that any program or operation that requires creating or writing FLEX files must do so on a "drive" or device other than FlexSys, once FlexSys has been made sharable.

1-3: Nomenclature

O-FLEX is the name of the package described herein, which consists of an executable OS-9 module "OFlex", and an OS-9 data file "FlexSys" (which is formatted as a FLEX disk) that contains the FLEX disk operating system and utilities which run under FLEX in the environment provided by the program OFlex.

OFlex provides disk drivers for FLEX which are capable of reading and writing floppy disks by using OS-9 system calls and Edition 10 or later of the device driver "G68".

A FLEX "drive" is referenced in the normal FLEX manner using drive numbers (0,1,2, or 3) and may refer to either a file on an OS-9 format disk that is internally formatted as a FLEX disk (using the O-FLEX command "OFORM", or to an entire floppy disk that is formatted with the standard FLEX format. Note: O-FLEX includes no provisions for formatting FLEX format floppy disks; they must be formatted on a stand-alone FLEX system.

SECTION 2: RUNNING O-FLEX

2-1:Command Syntax

The syntax of the OS-9 program OFlex is:

```
OFlex [0=<path>] [1=<path>] [2=<path>] [3=<path>] [P=<path>]
```

Where the numbers 0...3 refer to the paths that will be assigned to the respective FLEX drive numbers, and P is the path that will be used for printer output by the FLEX "P" command. The <path> given for a drive must be either a pathlist to an OS-9 file, structured as a FLEX disk (using OFORM.COMD) or the name of an OS-9 device (floppy disk drive) containing a FLEX-format floppy disk, followed by "@". The <path> given for "drive" 0 must point to a "disk" that includes a copy of the O-FLEX version of the FLEX operating system, linked per standard FLEX convention, such as the supplied "FlexSys" file. The <path> given for P can be a pathlist to any OS-9 file or device that can be opened for write. Pathlists begin in the current DATA directory.

For Example:

```
OS9: OFlex 0=FlexSys 1=MyDisk 2=/d1@ P=/p1
```

This command would attempt to boot FLEX from the file "FlexSys" in the current data directory and, if successful, open paths to the file Mydisk (FLEX "drive" #1) and a FLEX format disk in drive D1 (FLEX "drive" #2) Printer output (FLEX P command) would be directed to the device P1. The default system and work drives (see the FLEX command "ASN") will be drive 0.

If OFlex is called without command line parameters, the following default parameters are used:

0=.../SYS/FlexSys	The supplied "system disk" file.
1=FlexData	A file structured as a FLEX "disk", user created with "OFORM"
P=FlexPrint	See the "Printer Access" section

FLEX "drives" that are not assigned explicitly, or by default, can be assigned later using the "ASNDISK" command. Attempts to access unassigned drives will return a "Drive Not Ready" error.

OFlex allocates 56K of static storage and verifies that it starts at absolute address \$0000. Following this, it attempts to open the path associated with drive 0 and boot FLEX. Standard conventions for linking FLEX.SYS apply.

After reading the file linked on drive 0, OFlex sets up an interface area beginning at absolute address \$DE00. This area is essentially a jump table for access to drivers provided within OFlex for disk, terminal, and printer I/O. OFlex then jumps to the file loaded (FLEX.SYS). At this point, FLEX performs its own initialization and issues the normal FLEX prompt (+++).

2-2:Returning to OS-9

OFlex replaces the built-in FLEX command "MON" with the command "OS9". The "OS9" command terminates FLEX and returns to OS-9, restoring appropriate path options and closing any open paths to disks or files. This command is the only safe way to exit FLEX and return to OS-9. An abnormal exit may result in damage to FLEX files or disks, which are open for write operations.

2-3:Console I/O

Console I/O is vectored to Standard Input and Standard Output. The OS-9 I/O path options are saved and modified appropriately; they are restored to their former values on a normal exit from OFlex. Console output is buffered by OFlex for increased performance with I/O processors.

2-4:Floppy Disk Access

If the pathname given for a drive (0...3) ends with the character "@", OFlex assumes it is a path to a drive, containing a FLEX-format floppy disk. OFlex maintains a table of disk parameters (number of sides, density, and track density) which is used to set up the drive table in the OS-9 disk driver. OFlex computes the appropriate Logical Sector Number from the track and sector given, and attempts to access the disk.

If access errors occur, OFlex will retry with all possible combinations of disk parameters before returning an error. Since these retries are in addition to retries by OS-9, this is a time-consuming process. When a drive is opened, it is assumed to be 96 tpi (if the device descriptor allows), and double density. The parameters are tried in the following order on error: 96 tpi, SD; 48 tpi, DD; 48 tpi, SD; and 96 tpi, DD again. Note that 96 tpi selections are skipped if the device descriptor indicates other than an 80 track drive. Also, If a track number greater than 40 is requested on a five inch drive in 48 tpi mode, an error is automatically generated, forcing a retry.

2-5:File Access

If the pathname given for a drive (0...3) does not end with the character "@", OFlex assumes a file formatted like a FLEX floppy disk. Logical sector 0 of the file corresponds to Track 0, Sector 1. Logical Sector 2, the SIR, specifies the number of tracks and sectors on the disk; the file is assumed to contain exactly this number of total sectors. Note that track 0 contains the specified number of sectors, unlike the situation with double-density FLEX floppies. The Logical Sector Number is computed as follows:

$$\text{LSN} = \text{Track} * \text{Sectors_per_track} + \text{Sector} - 1$$

2-6:Record Locking

On multi-user systems, care must be taken to prevent one FLEX from reading or writing a disk or file that another FLEX is writing. For this reason, the following precautions are taken.

Raw-mode drives (entire disks) are file-locked when they are opened. No sharing of floppy disks is allowed. If a disk is already opened when FLEX tries to open it, a "DRIVE NOT READY" error will be returned. Note that this technique does not prevent another user from attempting to open an OS-9 file on the disk.

Data-file drives (OS-9 files) may be written only if the non-sharable attribute is set in the OS-9 file descriptor. Otherwise, a write-protect error will be returned. The non-sharable (-s-----) bit may be thought of as a substitute for a write-protect tab.

Note: Flex will not attempt to use the default pathname (i.e. "1=FlexData") more than once. If an error is returned because the file is non-sharable and busy or the file does not exist, a pathname must be given with the ASNDISK command to assign the drive.

2-7:Printer Access

Printer output is handled essentially as described in the FLEX User's Manual. O-FLEX includes the printer driver "PRINT.SYS" that calls a subroutine "POUT" in OFlex.

Calls to POUT are directed to the path given with the "P=" option on the OFlex command line. The path specified may be either a disk file (OS-9 format), or a device (/p, /p1, /ip1, etc.). By default, this path is a file, FlexPrint, in the current data directory. On the first request for printer output, OFlex attempts to open a path to the file or device specified. If successful, the output, and all subsequent output requests are directed to this path. Note: If the path opened is an existing file, its size is set to zero, effectively deleting the current contents of the file. If the attempt to open the path fails (because the file or device does not exist), OFlex attempts

to create a file using the specified name. If no valid path can be opened, OFlex directs the output to the standard output (console) path.

Printer output saved in a file can be printed later under OS-9 or, if a path to a printer is specified, printed directly from FLEX in the normal manner. NOTE: When a path to a printer or print file is opened, the path remains open until FLEX terminates. Since printers are normally non-sharable devices, this means that the printer will be unavailable to other users. For this reason, it may be preferable to direct the "printer" output to a file for later printout under OS-9.

When printer output is directed to an OS-9 file for later printout, it contains an exact image of the data that would have been sent to the printer. Since this image includes both carriage return and line feed characters, using the OS-9 "List" command to list the file will normally cause the addition of an extra line feed for each carriage return. To avoid this problem, use the OS-9 "Merge" command in place of "List" to examine or print O-FLEX printer output files. For example: "Merge Flexprint" to list the file on the console or "Merge FlexPrint >/P" to list the file on a printer.

2-8:Memory Access

OFlex allocates 56K of static storage and will report an error if this storage does not begin at absolute address \$0000. The area from \$DE00 to \$DFFF is used to interface with FLEX and should not be written. No absolute addresses are known above \$DFFF, and attempts to access them will yield unpredictable results, with one exception. On Gimix III systems, writing to any address above \$DFFF, and reading any address in this range that does not lie in the OFlex module, will result in either "ERROR 198" or "ERROR 199" and OFlex will terminate, returning the user to OS-9. The Standard I/O path options will have the altered values set by OFlex (this will usually result in a loss of input echo which can be restored by using: TMODE echo) and, if a FLEX file was open for write, corruption of the FLEX disk is likely.

2-9:Interrupts

Since hardware and software interrupt processing is handled by OS-9, FLEX software that uses interrupts will not work with O-FLEX. In addition, programs should not mask interrupts. On GMX III systems, masking of interrupts can result in a hardware "trap" and cause O-FLEX to abort. On GMX II systems, masking of the interrupts by FLEX programs can prevent other users on the system from being serviced.

SECTION 3: O-FLEX FILES and UTILITIES

This section documents the O-FLEX specific files included on "FlexSys". Additional files, the standard FLEX command set, plus some additional commands provided by GIMIX, are documented in the "FLEX USER'S MANUAL" included with O-FLEX. Note: The "FLEX USER'S MANUAL" contains some information on system requirements and configuration which does not apply to O-FLEX. O-FLEX does NOT require any changes in the configuration of the OS-9 system on which it is used.

3-1:FLEX.SYS

The FLEX.SYS provided is setup to run in the O-FLEX environment. Besides the obvious changes to link to the disk drivers, et cetera, some initialization code has been added. This code gets the date from OS-9 and sets the date register. Also, certain TTYSET parameters are initialized based on the Standard I/O path options.

3-2:XBASPAT.TXT (.BIN)

These files contain source and object code for the patches necessary to permit Technical Systems Consultant's XBASIC to run with O-FLEX. When the .BIN file is appended to XBASIC it sets up the terminal status routine and changes the vector for the "EXIT" command to the FLEX warmstart entry.

3-3:ASMBPATO.TXT (.BIN)

Another patch file, this one is appended to the Technical Systems Consultant's Assembler (ASMB) to print the time on the page banner. Note: This patch is an enhancement to the assembler and is not required for operation under O-FLEX. The patch is written for Version #2 of the assembler and may not work with other versions.

3-4:OFLEX.DEF

This is an assembler "LIB" file containing the definitions for the OFlex interface area beginning at \$DE00. OFlex replaces the blank 3-byte entries in this area with jump instructions to the appropriate routines within itself. This file is used when assembling the included source files.

The first 10 entries in this area are the jumps to the disk drivers as described in the FLEX manuals. The remaining entries are special entries directly into OFlex and user written programs or patches should not need to use them. To retain compatibility with other versions of FLEX, use the standard entry points described in the FLEX manuals.

3-5:OFORM.CMD

Its function is to create a new OS-9 file which is structured as a FLEX disk and is usable by O-FLEX. The file must not already exist. It is created non-sharable and with no public permissions.

The general syntax of the OFORM command is:

```
OFORM <pathname> <vol. name>[.<ext>] <vol. #> <tracks> <sectors>
```

<pathname> is a standard OS-9 path list, with the last name being a file name for the "disk" to be created. <vol. name>, <.ext> and <vol. #> follow the standard FLEX conventions for disk names. <tracks> and <sectors> (in decimal) determine the size of the resulting O-FLEX "disk". The number of tracks must be at least two and at most 255. The number of sectors must be at least 5 and at most 255. Note that the resulting "disk" will have <tracks>*<sectors> sectors in it.

Example: OFORM TempDisk TEMP.DSK 1 40 20

3-6:SHELL.CMD

This command forks an OS-9 shell and passes a command to it. The path options are temporarily restored. If <command> is omitted or is a built-in shell command such as "chd", the user will remain in the "SHELL" and control will not return to FLEX until an EOF character (ESC) is received by the shell.

The general syntax of the SHELL command is:

```
SHELL <command>
```

Example: SHELL dir e

3-7:ASNDISK.CMD

The function of the ASNDISK command is to associate a FLEX drive number with a new or different OS-9 path. The same rules apply here that apply to the options on the command line. The old path associated with <drive> is closed, if open. <path> may be left blank to close a path without opening a new one.

The general syntax of the ASNDISK command is:

```
ASNDISK <drive>=<path> [<drive>=<path> ...]
```

Example: ASNDISK 1=TempDisk 2=/d1@ 3=

In the above example, "drive" 1 is assigned to the file TempDisk, 2 is assigned to the FLEX format floppy disk in device /d1, and "drive" 3 is de-assigned; any path previously associated with it is closed.

3-8:XCOPY.COMD

XCOPY takes advantage of having access to both OS-9 and FLEX to provide a file transfer utility between OS-9 and FLEX. If <src-file> or <dest-file> begins with the string "OS9%", the remainder of the string will be used as an OS-9 pathname, otherwise a standard FLEX file spec is assumed.

The general syntax of the XCOPY command is:

```
XCOPY <src-file> <dest-file> [+B]
```

The +B option indicates that the FLEX file(s) are binary, not text. In either case, Flex files default to .TXT extension and the work drive.

XCOPY uses single-character I/O and is slower than the normal FLEX or OS-9 copy commands; which should be used when copying like file types.

```
Examples: XCOPY os9%letter 2.LETTER.TXT
          XCOPY 0.FLEX.SYS os9%FlexSys +B
```

The first example copies the OS-9 format file "letter" to a FLEX text file on "drive" 2. The second copies 0.FLEX.SYS to an OS-9 format file "FlexSys" in the current data directory, in binary format. Note: When FLEX text-format files (w/space compression) are copied to OS-9 files, XCOPY performs the necessary space de-compression to produce a standard OS-9 format text file.

3-9:TIME.COMD (.TXT)

This command displays the current date and time from the system time-of-day clock. The source for this program provided as a guide to accessing the OS-9 time information from FLEX programs.

The general syntax of the TIME command is:

```
TIME
```

SECTION 4: SOFTWARE COMPATIBILITY

4-1: Incompatible Utilities

The following commands, normally included with FLEX and/or GIMIX FLEX, are NOT supplied with O-FLEX. These commands are either incompatible with O-FLEX or perform functions that are provided by standard OS-9 utilities or functions.

BACKUP	CLEAN	FORMAT
FIXDAY(3)	FIXDAY3	HARDFxxx
PRINT.COMD	QCHECK	REPORT
SETTIME	SETUP	SYSGEN
TIME3	YEAR	

4-2: FLEX Diagnostics Package

The memory diagnostics in the Technical Systems Consultants "FLEX Diagnostics" package cannot be used with O-FLEX. These programs make I/O calls to the "system monitor" which does not exist in O-FLEX.

Some of the disk diagnostics in existing versions of this package do not support non-standard disk formats and will not work with OS-9 files formatted as FLEX disks. Updated versions that accept non-standard disk formats will be available from Technical Systems Consultants. Contact them for information on updates.

4-3: FLEX Extended BASIC (XBASIC)

Source and object for the necessary patches to Technical Systems Consultants' Extended BASIC are included with O-FLEX. See section 3-2.

4-4: FLEX Debug Package

Current versions of Technical Systems Consultants' "FLEX Debug" package are non-interruptable and will not work with O-FLEX.

4-5: Compatibility with Other Software

Much of the software written for other versions of FLEX should work with O-FLEX. However, there are certain restrictions and exceptions. In some cases, it may be possible to modify the software to permit its use with O-FLEX.

***** CAUTION *****

On OS-9 GMX III systems, attempts to use software that violates the restrictions described in the following paragraphs will probably result in a hardware trap error (Error #197, #198, or #199) and only cause an abnormal exit from FLEX. On GMX II systems; however, the results will generally be unpredictable and may crash the entire system. For this reason it is advisable to test untried software on a quiet system to prevent loss of data or damage to other users files.

Software that directly accesses hardware such as I/O boards, memory-mapped video displays, EPROM programmers, memory-management hardware, the time-of-day clock, etc. will NOT work with O-FLEX. These devices are not mapped into the O-FLEX address space by OS-9 and are therefore inaccessible. Some software, like XBASIC, directly accesses the terminal I/O port when checking for keyboard input. This can usually be corrected by patching the software to use the "STAT" call in FLEX (See the section on XBASPAT and the section on patching software for O-FLEX).

All I/O must use the standard FLEX I/O system calls described in the FLEX PROGRAMMER'S MANUAL. Since the usual "system monitor" program (GMXBUG, etc.) is not present, software that uses monitor calls for I/O must be modified to use FLEX calls instead. Software that uses monitor calls for other functions may need to be re-written to include the necessary functions internally.

Software that uses interrupts (like the FLEX Print Spooler) will NOT work with O-FLEX. This restriction includes both hardware and software interrupts, since FLEX has no control over the handling of interrupts or the interrupt vectors. In addition, the software must not mask interrupts, even if it does not actually use them. Masking of the interrupts can prevent other users on a multi-user system from being serviced and, in GMX III systems, is likely to cause FLEX to be aborted by a hardware (Watchdog) trap.

4-6: Patching Software to Run with O-FLEX

It is difficult to give specific information on software patching; however, the most likely problems will be programs that either use system monitor calls or directly access hardware for I/O functions. This type of problem can often be corrected by changing the software so that it uses standard FLEX I/O calls to perform these functions. The source code provided for the XBASIC and ASMB patches, and for the TIME command, can be used as a guide to correcting certain problems in other software.

