

```

EEEEEEEEEE XX XX AAAAAA MM MM PPPPPPPP LL EEEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AAAAAA MM MM PPPPPPPP LL EEEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AAAAAA MM MM PPPPPPPP LL EEEEEEEEEE SSSSSSSS
EE XX XX AA AA MMMM MMMM PP PP LL EE SS
EE XX XX AA AA MMMM MMMM PP PP LL EE SS
EE XX XX AA AA MMMM MMMM PP PP LL EE SS
EE XX XX AA AA MM MM MM PP PP LL EE SS
EE XX XX AA AA MM MM MM PP PP LL EE SS
EE XX XX AA AA MM MM MM PP PP LL EE SS
EEEEEEEEEE XX XX AA AA MM MM PPPPPPPP LL EEEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AA AA MM MM PPPPPPPP LL EEEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AA AA MM MM PPPPPPPP LL EEEEEEEEEE SSSSSSSS
EE XX XX AAAAAAAAAA MM MM PP LL EE SS
EE XX XX AAAAAAAAAA MM MM PP LL EE SS
EE XX XX AAAAAAAAAA MM MM PP LL EE SS
EE XX XX AA AA MM MM PP LL EE SS
EE XX XX AA AA MM MM PP LL EE SS
EE XX XX AA AA MM MM PP LL EE SS
EEEEEEEEEE XX XX AA AA MM MM PP LLLLLLLLLL EEEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AA AA MM MM PP LLLLLLLLLL EEEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AA AA MM MM PP LLLLLLLLLL EEEEEEEEEE SSSSSSSS

```

```

GGGGGGGG  BBBB8888  LL      SSSSSSSS  EEEEEEEEEE  CCCCCCCC  UU      UU  FFFFFFFF  000000
GGGGGGGG  BBBB8888  LL      SSSSSSSS  EEEEEEEEEE  CCCCCCCC  UU      UU  FFFFFFFF  000000
GG      BB      BB  LL      SS      EE      CC      UU      UU  FF      00      00
GG      BB      BB  LL      SS      EE      CC      UU      UU  FF      00      00
GG      BB      BB  LL      SS      EE      CC      UU      UU  FF      00      00
GG      BB      BB  LL      SS      EE      CC      UU      UU  FF      00      00
GG      BBBB8888  LL      SSSSSS  EEEEEEEC  CC      UU      UU  FFFFFFFF  00      00
GG      BBBB8888  LL      SSSSSS  EEEEEEEE  CC      UU      UU  FFFFFFFF  00      00
GG  GGGGGG  BB      BB  LL      SS      EE      CC      UU      UU  FF      00      00
GG  GGGGGG  BB      BB  LL      SS      EE      CC      UU      UU  FF      00      00
GG      GG  BB      BB  LL      SS      EE      CC      UU      UU  FF      00      00
GG      GG  BB      BB  LL      SS      EE      CC      UU      UU  FF      00      00
GG      GG  BB      BB  LL      SS      EE      CC      UU      UU  FF      00      00
GGGGGG  BBBB8888  LLLLLLLLLL  SSSSSSSS  EEEEEEEEEE  CCCCCCCC  UUUUUUUUU  FF      000000  .....
GGGGGG  BBBB8888  LLLLLLLLLL  SSSSSSSS  EEEEEEEEEE  CCCCCCCC  UUUUUUUUU  FF      000000  .....

```

```

MM      MM  AAAAAA  RRRRRRRR
MM      MM  AAAAAA  RRRRRRRR
MMMM  MMMM  AA      AA  RR      RR
MMMM  MMMM  AA      AA  RR      RR
MM  MM  MM  AA      AA  RR      RR
MM  MM  MM  AA      AA  RR      RR
MM      MM  AA      AA  RRRRRRRR
MM      MM  AA      AA  RRRRRRRR
MM      MM  AAAAAAAAAA  RR  RR
MM      MM  AAAAAAAAAA  RR  RR
MM      MM  AA      AA  RR      RR
MM      MM  AA      AA  RR      RR
MM      MM  AA      AA  RR      RR
M 1      MM  AA      AA  RR      RR

```

LAE

.SE

••  
••  
••  
••

AD.

AD.

AD.

AD.

AD.

AD.

AD.

AD.

••

AD.

••

••

••

••

AD.

.TITLE GBLSECUFO  
 .IDENT 'V04-000'

Global Section UFO (User File Open)

```

*****
*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*
* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*
* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*
* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****
  
```

```

:This routine opens a file to be used as a global section
:An RMS OPEN is performed with the file options (FOP) of
:User File Open (UFO). The calling routine specifies the
:file name and number of blocks; this routine returns the
:channel number on which the file was opened.
:If the specified file does not exist, the file is created
  
```

```

:The calling sequence is
  
```

```

    Call GBL_SECTION_UFO( blkcnt,file-name,chan )
  
```

```

:Where
  
```

```

    blkcnt => Number of blocks in the file
    file-name => filename descriptor block
    chan => channel opened
  
```

```

:Example:
  
```

```

    Integer*4 CHANNEL
    :
    Call GBL_SECTION_UFO(10,'LABIO_DATA.DAT',CHANNEL )
  
```

```
.SBTTL  GBL_SEC_UFO
; RMS FAB for a $CREATE
GBLFAB: $FAB  FAC=PUT,-
             FOP=<UFO,CIF,CBT>

NUM_ARG = 3           ;Number of arguments

.ENTRY  GBL_SECTION_UFO,0

MOVL   #SS$_INSFARG,R0      ;Assume bad arg count
CMPB   (AP),#NUM_ARG       ;Check arg count
BLSS   EXIT                ;Too few

MOVL   8(AP),R1             ;Get file name address string descriptor
MOVB   (R1),GBLFAB+FAB$B_FNS ;Store string length in FAB
MOVL   4(R1),GBLFAB+FAB$_FNA ;And file name

MOVL   @4(AP),GBLFAB+FAB$_ALQ ;Number of blocks to allocate

$CREATE FAB=GBLFAB         ;Open data file, Create it if
                           ;if it does not exist
MOVL   GBLFAB+FAB$_STV,@12(AP);Store channel number

EXIT:   RET                ;Return with error code in R)

.END
```

LPMULT B32

DRMST MAR

ADDRIVER MAR

TORIVER MAR

USSTEST MAR

GBLSECUR MAR

USSDISP MAR

DOD\_ERAPAT MAR

LBRMAC MAR

XADDRIVER MAR

LABLOCIN MAR

DRSLU MAR

DTE\_DF03 MAR

SECRET MAR

WORKO LIS

EXAMPLES