



```

RRRRRRRR      MM      MM      SSSSSSSS      SSSSSSSS      HH      HH      RRRRRRRR
RRRRRRRR      MM      MM      SSSSSSSS      SSSSSSSS      HH      HH      RRRRRRRR
RR      RR      MMMM      MMMM      SS      SS      HH      HH      RR      RR
RR      RR      MMMM      MMMM      SS      SS      HH      HH      RR      RR
RR      RR      MM      MM      SS      SS      HH      HH      RR      RR
RRRRRRRR      MM      MM      SSSSSS      SSSSSS      HHHHHHHHHH      RRRRRRRR
RRRRRRRR      MM      MM      SSSSSS      SSSSSS      HHHHHHHHHH      RRRRRRRR
RR      RR      MM      MM      SS      SS      HH      HH      RR      RR
RR      RR      MM      MM      SS      SS      HH      HH      RR      RR
RR      RR      MM      MM      SS      SS      HH      HH      RR      RR
RR      RR      MM      MM      SSSSSSSS      SSSSSSSS      HH      HH      RR      RR
RR      RR      MM      MM      SSSSSSSS      SSSSSSSS      HH      HH      RR      RR

```

```

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

```

```

SSSSSSSS      DDDDDDDD      LL
SSSSSSSS      DDDDDDDD      LL
SS      DD      DD      LL
SS      DD      DD      LL
SS      DD      DD      LL
SS      DD      DD      LL
SSSSSS      DD      DD      LL
SSSSSS      DD      DD      LL
SS      DD      DD      LL
SS      DD      DD      LL
SS      DD      DD      LL
SS      DD      DD      LL
SSSSSSSS      DDDDDDDD      LLLLLLLLLL
SSSSSSSS      DDDDDDDD      LLLLLLLLLL

```

RM  
C  
C  
C  
mo  
/\*  
/\*  
/\*  
ag

{ \$begin rmsshr,V04-000

```

{*****
{*
{* 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.
{*
{*****
{

```

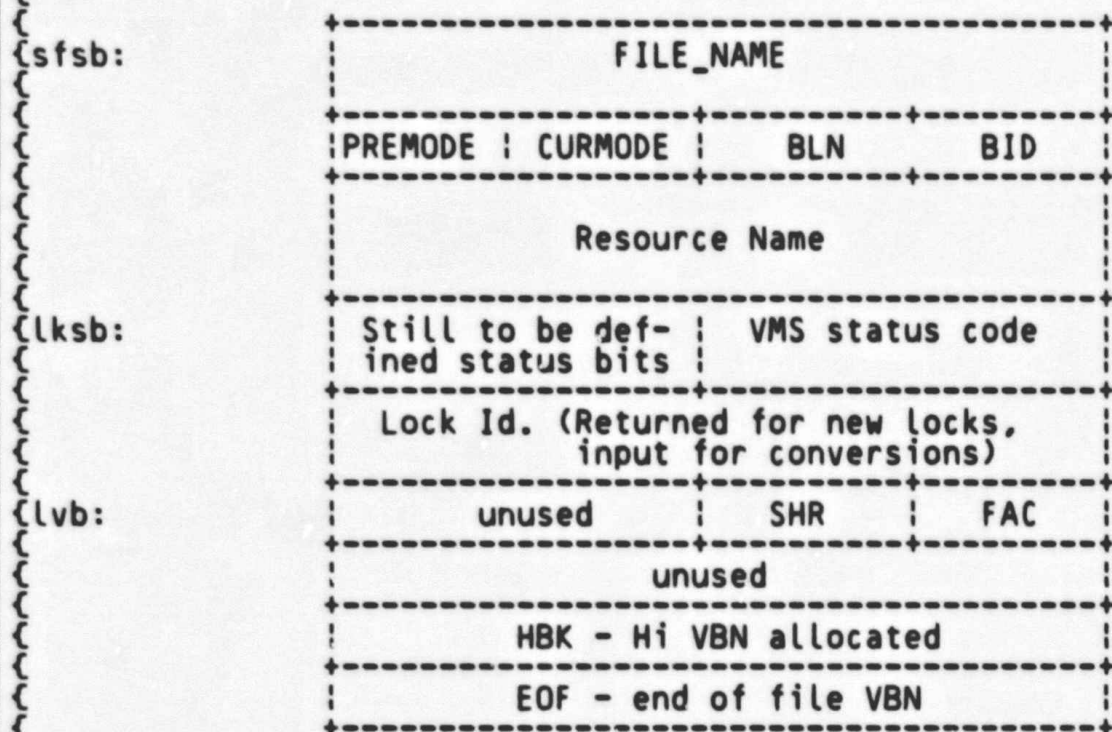
## rms shared file structure definitions

## Modified By:

V03-009	SHZ0005	Stephen H. Zalewski	05-Aug-1983	
	Add a one byte flag field to GBSB.			
V03-008	SHZ0004	Stephen H. Zalewski	28-Jun-1983	
	Change size of RMS facility code in SFSB.			
V03-007	KBT0489	Keith B. Thompson	9-Feb-1983	
	Add FAB and SHR field to sfsb			
V03-006	KBT0325	Keith B. Thompson	10-Sep-1982	
	Remove all SO sharing structure definitions			
V03-005	SHZ0003	Stephen H. Zalewski,	12-Aug-1982	14:09
	Added further definitions to GBSB.			
V03-004	SHZ0002	Stephen H. Zalewski,	11-Aug-1982	21:10
	Changed name of \$GBLB to \$GBSB.			
V03-003	SHZ0001	Stephen H. Zalewski	10-Aug-1982	
	Add GBLB definitions.			
V03-002	KBT0081	Keith B. Thompson	12-Jul-1982	
	Add b_curlck, l_ebk and l_hbk to the sfsb. Also fix revision number in jwh000T			
V03-001	JWH0001	Jeffrey W. Horn	24-Mar-1982	
	Change SIFB to reflect journaling changes to the IFB.			

SFSB field definitions - shared file synchronization block

The SFSB contains the information necessary to lock a file for RMS synchronization via the VAX/VMS Lock Manager.



module \$SFSBDEF;

aggregate SFSBDEF structure fill prefix SFSB\$;

FILENAME OVERLAY union fill;  
 FILENAME quadword unsigned;

/\* descriptor of shared file resource name.  
 /\* resource name is NODE, DEVICE, FILE\_ID  
 /\* points to RESNAM, below

FILENAME\_FIELDS structure fill;  
 NAME\_LEN word unsigned;  
 FILL\_3 byte dimension 2 fill prefix SFSBDEF tag \$\$;  
 ADDRESS longword unsigned;

/\* subfield to address descriptor length field  
 /\* subfield to address descriptor address field

end FILENAME\_FIELDS;  
 end FILENAME\_OVERLAY;

BID byte unsigned;  
 constant BID equals 16 prefix SFSB tag \$C;

/\* block id  
 /\* sfsb code  
 /\* block length in longwords  
 /\* Mode of the current lock  
 /\* Mode of the previous lock

BLN byte unsigned;  
 CURMODE byte unsigned;  
 PREMODE byte unsigned;

RESNAM OVERLAY union fill;  
 RESNAM character length 32;  
 RESNAM\_FIELDS structure fill;  
 FAC\_CODE longword unsigned;  
 FID\_NUM word unsigned;

/\* 32 bytes for name of shared resource  
 /\* RMS facility code (RMSS)  
 /\* file id word one

```

      FID_SEQ word unsigned;          /* file id word two
      FID_RVN word unsigned;         /* file id word three
      constant FIX_LEN equals 10 prefix SFSB tag $C; /* 10 bytes of fixed size data
      DEV_NAM character length 22;   /* 22 bytes remain to hold device id (node$device_name)
    end RESNAM_FIELDS;
  end RESNAM_OVERLAY;
  LKSB_OVERLAY union fill;
    LKSB longword unsigned;          /* lock status block
    LKSB_FIELDS structure fill;
      STATUS word unsigned;         /* VMS status code
      S_BITS word unsigned;         /* various status bits
    end LKSB_FIELDS;
  end LKSB_OVERLAY;
  LOCK_ID longword unsigned;         /* second longword of LKSB is the lock id
  LVB_OVERLAY union fill;
    LVB longword unsigned dimension 4; /* lock value block
    constant BLN equals . prefix SFSB$ tag K; /* length of sfsb
    constant BLN equals . prefix SFSB$ tag C; /* length of sfsb
/*
/*
/*
    keep the next two fields in same order as they are in FAB
  LVB_FIELDS structure fill;
    FAC byte unsigned;              /* fac bits from FAB
    SHR byte unsigned;              /* sharing bits (from FAB SHR field)
    FILL_1 word fill prefix SFSBDEF tag $$; /* spare
    FILL_2 longword fill prefix SFSBDEF tag $$; /* spare
    HBK longword unsigned;          /* high block
    EBK longword unsigned;          /* end of file
  end LVB_FIELDS;
end LVB_OVERLAY;
end SFSBDEF;

end_module $$SFSBDEF;

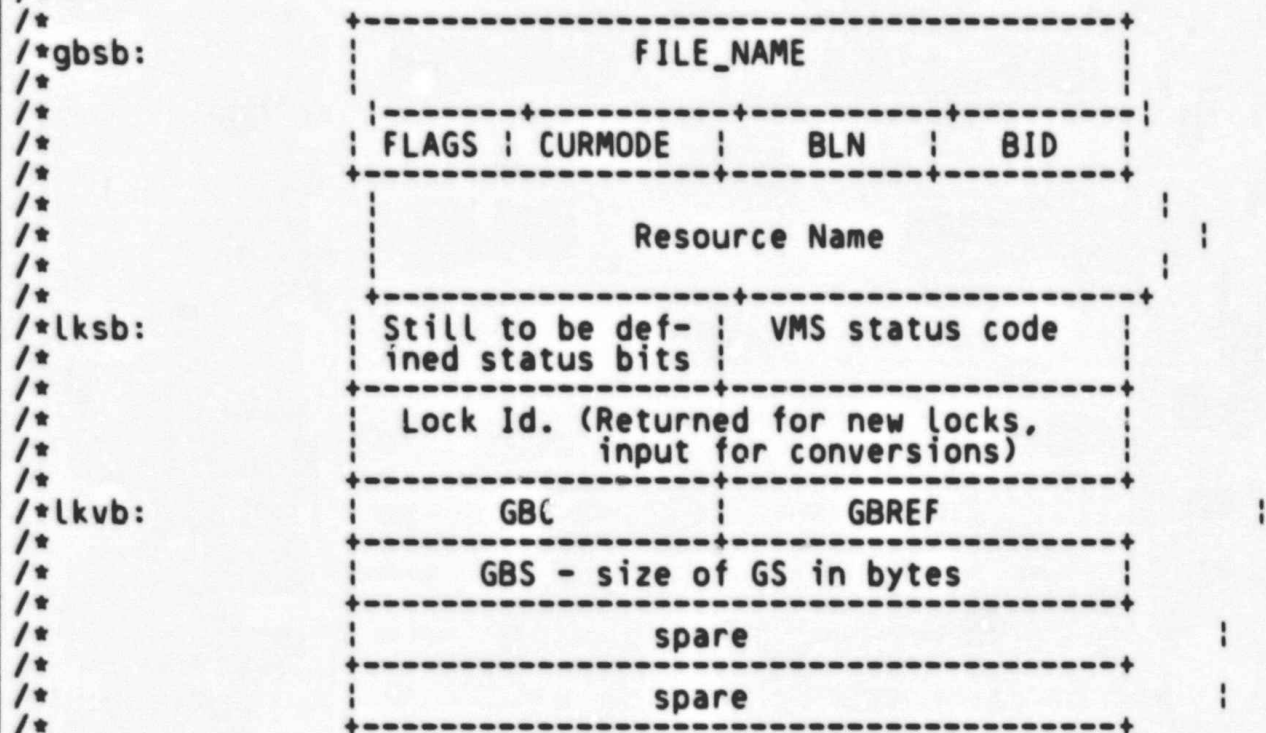
module $GBSBDEF;

```

```

/*
/*      GBSB field definitions - global buffer synchronization block
/*
/*      The GBSB contains the information necessary to determine if a
/*      global section is already open for a file on a given node, and
/*      is used for synchronizing access to the global section.
/*

```



```

aggregate GBSBDEF structure fill prefix GBSB$:
  FILENAME OVERLAY union fill;
    FILENAME quadword unsigned;
    FILENAME_FIELDS structure fill;
      NAME_LEN word unsigned;
      FILL_3 byte dimension 2 fill prefix GBSBDEF tag $$;
      ADDRESS longword unsigned;
    end FILENAME_FIELDS;
  end FILENAME_OVERLAY;
  BID byte unsigned;
  constant BID equals 9 prefix GBSB tag $C;
  BLN byte unsigned;
  CURMODE byte unsigned;
  FLAGS OVERLAY union fill;
    FLAGS byte unsigned;
    FLAGS_BITS structure fill;
      NOTACCESSED bitfield mask;
    end FLAGS_BITS;
  end FLAGS_OVERLAY;
  /* descriptor of shared file resource name.
  /* resource name is NODE, DEVICE, FILE_ID
  /* points to RESNAM, below
  /* subfield to address descriptor length field
  /* subfield to address descriptor address field
  /* block id
  /* gbsb code
  /* block length in longwords
  /* Mode of the current lock
  /* spare
  /* Process has already decremeted access count for GBS.

```

RMSI  
 /\*--  
 end  
 end.

```

RESNAM character length 32;          /* 32 bytes for name of shared resource
LKSB_OVERLAY union fill;
  LKSB longword unsigned;          /* lock status block
  LKSB_FIELDS structure fill;
    STATUS word unsigned;         /* VMS status code
    S_BITS word unsigned;        /* various status bits
  end LKSB_FIELDS;
end LKSB_OVERLAY;
LOCK_ID longword unsigned;        /* second longword of LKSB is the lock id
GBC word unsigned;               /* Number of global buffers in section.
GBREF word unsigned;            /* Number of accessors to global section.
GS_SIZE longword unsigned;      /* Size of global section in bytes.
FICL_1 longword fill prefix GBSBDEF tag $$; /* spare
FILL_2 longword fill prefix GBSBDEF tag $$; /* spare
constant BLN equals . prefix GBSB$ tag K; /* length of gbsb
constant BLN equals . prefix GBSB$ tag C; /* length of gbsb
end GBSBDEF;
end_module $GBSBDEF;

```



This image displays a grid of 120 small technical diagrams and code snippets, arranged in 10 rows and 12 columns. Each cell contains a different diagram or code block, many of which are labeled with titles such as RMSFILSTR, RMSINTSTR, RMSUSR, RMSMAC, RMSWADEF, and RMSSHR. The diagrams include various types of charts, flowcharts, and code listings, representing different components or configurations of the VAX/VMS system.