

Translation of VMS Function prototypes for system and layered product API's

PASCAL	C++
<pre> MODULE QA_FUNCTION_PROTOTYPES (input, output); [HIDDEN] TYPE \$BYTE = [BYTE] -128..127; \$WORD = [WORD] -32768..32767; [HIDDEN] TYPE \$BYTE = [BYTE] -128..127; \$WORD = [WORD] -32768..32767; \$QUAD = [QUAD,UNSAFE] RECORD L0:UNSIGNED; L1:INTEGER; END; \$OCTA = [OCTA,UNSAFE] RECORD L0,L1,L2:UNSIGNED; L3:INTEGER; END; \$SUBYTE = [BYTE] 0..255; \$UWORD = [WORD] 0..65535; \$UQUAD = [QUAD,UNSAFE] RECORD L0,L1:UNSIGNED; END; \$UOCTA = [OCTA,UNSAFE] RECORD L0,L1,L2,L3:UNSIGNED; END; \$PACKED_DEC = [BIT(4),UNSAFE] 0..15; \$DEFTYP = [UNSAFE] INTEGER; \$DEFPTR = [UNSAFE] ^\$DEFTYP; \$BOOL = [BIT(1),UNSAFE] BOOLEAN; \$BIT = [BIT(1),UNSAFE] BOOLEAN; \$BIT2 = [BIT(2),UNSAFE] 0..3; \$BIT3 = [BIT(3),UNSAFE] 0..7; \$BIT4 = [BIT(4),UNSAFE] 0..15; \$BIT5 = [BIT(5),UNSAFE] 0..31; \$BIT6 = [BIT(6),UNSAFE] 0..63; \$BIT7 = [BIT(7),UNSAFE] 0..127; \$BIT8 = [BIT(8),UNSAFE] 0..255; \$BIT9 = [BIT(9),UNSAFE] 0..511; \$BIT10 = [BIT(10),UNSAFE] 0..1023; \$BIT11 = [BIT(11),UNSAFE] 0..2047; \$BIT12 = [BIT(12),UNSAFE] 0..4095; \$BIT13 = [BIT(13),UNSAFE] 0..8191; \$BIT14 = [BIT(14),UNSAFE] 0..16383; \$BIT15 = [BIT(15),UNSAFE] 0..32767; \$BIT16 = [BIT(16),UNSAFE] 0..65535; </pre>	<pre> /** @file */ // // Sector7 USA Inc, VMS PASCAL to C++, Version 7.2 // Build: 8.000 2017/09/16 23:06:29 (GMT) // VX/RT (c) 1985-2013 Sector7 USA Inc. // Translated: 2017-10-09 15:40:58 CDT // Input File: ../PascalQA/QA_FunctionPrototypes.pas // Output File: ../PascalQA/QA_FunctionPrototypes.cxx // #include "VxPascal.hxx" typedef char \$BYTE ; typedef int16_t \$WORD ; typedef _QUAD \$QUAD; typedef struct \$OCTA { InitializeRecord(\$OCTA) UNSIGNED L0 ; UNSIGNED L1 ; UNSIGNED L2 ; INTEGER L3 ; } \$OCTA; typedef uchar_t \$SUBYTE ; typedef uint16_t \$UWORD ; typedef _QUAD \$UQUAD; typedef struct \$UOCTA { InitializeRecord(\$UOCTA) UNSIGNED L0 ; UNSIGNED L1 ; UNSIGNED L2 ; UNSIGNED L3 ; } \$UOCTA; </pre>

```

$BIT17 = [BIT(17),UNSAFE] 0..131071;
$BIT18 = [BIT(18),UNSAFE] 0..262143;
$BIT19 = [BIT(19),UNSAFE] 0..524287;
$BIT20 = [BIT(20),UNSAFE] 0..1048575;
$BIT21 = [BIT(21),UNSAFE] 0..2097151;
$BIT22 = [BIT(22),UNSAFE] 0..4194303;
$BIT23 = [BIT(23),UNSAFE] 0..8388607;
$BIT24 = [BIT(24),UNSAFE] 0..16777215;
$BIT25 = [BIT(25),UNSAFE] 0..33554431;
$BIT26 = [BIT(26),UNSAFE] 0..67108863;
$BIT27 = [BIT(27),UNSAFE] 0..134217727;
$BIT28 = [BIT(28),UNSAFE] 0..268435455;
$BIT29 = [BIT(29),UNSAFE] 0..536870911;
$BIT30 = [BIT(30),UNSAFE] 0..1073741823;
$BIT31 = [BIT(31),UNSAFE] 0..2147483647;
$BIT32 = [BIT(32),UNSAFE] UNSIGNED;

```

[HIDDEN] TYPE (**** SDL-Generated type names ****)

```

lib$routines$$typ1 = ^$QUAD;
lib$routines$$typ2 = ^$QUAD;
lib$routines$$typ3 = ^$QUAD;
lib$routines$$typ4 = ^$QUAD;
lib$routines$$typ5 = ^$QUAD;
lib$routines$$typ6 = ^$QUAD;
lib$routines$$typ7 = ^$QUAD;
lib$routines$$typ8 = ^$QUAD;
lib$routines$$typ9 = ^$QUAD;
lib$routines$$typ10 = ^$QUAD;
lib$routines$$typ11 = ^$QUAD;
lib$routines$$typ12 = ^$QUAD;

```

```

d_float$$type = double;
g_float$$type = double;

```

```

} $UOCTA;

```

```

typedef uchar_t $PACKED_DEC ;
typedef INTEGER $DEFTYP ;
typedef $DEFTYP& $DEFPTR ;
typedef BOOLEAN $BOOL ;
typedef BOOLEAN $BIT ;
typedef uchar_t $BIT2 ;
typedef uchar_t $BIT3 ;
typedef uchar_t $BIT4 ;
typedef uchar_t $BIT5 ;
typedef uchar_t $BIT6 ;
typedef uchar_t $BIT7 ;
typedef uchar_t $BIT8 ;
typedef uint16_t $BIT9 ;
typedef uint16_t $BIT10 ;
typedef uint16_t $BIT11 ;
typedef uint16_t $BIT12 ;
typedef uint16_t $BIT13 ;
typedef uint16_t $BIT14 ;
typedef uint16_t $BIT15 ;
typedef uint16_t $BIT16 ;
typedef uint32_t $BIT17 ;
typedef uint32_t $BIT18 ;
typedef uint32_t $BIT19 ;
typedef uint32_t $BIT20 ;
typedef uint32_t $BIT21 ;
typedef uint32_t $BIT22 ;
typedef uint32_t $BIT23 ;
typedef uint32_t $BIT24 ;
typedef uint32_t $BIT25 ;
typedef uint32_t $BIT26 ;
typedef uint32_t $BIT27 ;
typedef uint32_t $BIT28 ;
typedef uint32_t $BIT29 ;
typedef uint32_t $BIT30 ;
typedef uint32_t $BIT31 ;
typedef UNSIGNED $BIT32 ;

```

```

//! @ingroup QA_FUNCTION_PROTOTYPES
/** SDL-Generated type names **

```

```

typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP1 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP2 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP3 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP4 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP5 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP6 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP7 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP8 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP9 ;

```

```
[ASYNCHRONOUS] FUNCTION smg$put_chars (
    display_id : UNSIGNED;
    text : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER] OF CHAR;
    start_row : INTEGER := %IMMED 0;
    start_column : INTEGER := %IMMED 0;
    flags : UNSIGNED := %IMMED 0;
    rendition_set : UNSIGNED := %IMMED 0;
    rendition_complement : UNSIGNED := %IMMED 0;
    character_set : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;
```

```
[ASYNCHRONOUS] FUNCTION lib$create_vm_zone_64 (
    VAR zone_id : [VOLATILE] $UQUAD;
    algorithm : $QUAD := %IMMED 0;
    algorithm_argument : $QUAD := %IMMED 0;
    flags : $UQUAD := %IMMED 0;
    extend_size : $QUAD := %IMMED 0;
    initial_size : $QUAD := %IMMED 0;
    block_size : $QUAD := %IMMED 0;
    alignment : $QUAD := %IMMED 0;
    page_limit : $QUAD := %IMMED 0;
    smallest_block_size : $QUAD := %IMMED 0;
    zone_name : [CLASS_S] PACKED ARRAY [$111..$u11:INTEGER] OF CHAR
    := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE get_page := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE free_page := %IMMED 0) :
    INTEGER; EXTERNAL;
```

```
[ ASYNCHRONOUS] PROCEDURE OTS$MOVE3
(%IMMED len : INTEGER;
%REF src_str: [UNSAFE] PACKED ARRAY [11..h1: INTEGER] OF CHAR;
%REF dst_str: [UNSAFE] PACKED ARRAY [12..h2: INTEGER] OF CHAR
); EXTERNAL;
```

```
[ASYNCHRONOUS] FUNCTION lib$addx (
    %REF addend_array : [UNSAFE] ARRAY [$11..$u1:INTEGER] OF
    $SUBYTE;
```

```
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP10 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP11 ;
typedef _Ref<struct $QUAD >LIB$ROUTINES$$TYP12 ;
typedef DOUBLE D_FLOAT$$TYPE ;
typedef DOUBLE G_FLOAT$$TYPE ;
```

```
extern "C" int32_t SMG$PUT_CHARS(
/* |VR| */ /* uint32_t* DISPLAY_ID ,
/* |VR| %C_S| /* DXX& TEXT ,
/* |VR| /* int32_t* START_ROW _Imm(0),
/* |VR| /* int32_t* START_COLUMN _Imm(0),
/* |VR| /* uint32_t* FLAGS _Imm(0),
/* |VR| /* uint32_t* RENDITION_SET _Imm(0),
/* |VR| /* uint32_t* RENDITION_COMPLEMENT
_Imm(0),
/* |VR| /* uint32_t* CHARACTER_SET _Imm(0)) ;
```

```
extern "C" int32_t LIB$CREATE_VM_ZONE_64(
/* |VR| VL| /* void* ZONE_ID ,
/* |VR| /* void* ALGORITHM _Imm(0),
/* |VR| /* void* ALGORITHM_ARGUMENT _Imm(0),
/* |VR| /* void* FLAGS _Imm(0),
/* |VR| /* void* EXTEND_SIZE _Imm(0),
/* |VR| /* void* INITIAL_SIZE _Imm(0),
/* |VR| /* void* BLOCK_SIZE _Imm(0),
/* |VR| /* void* ALIGNMENT _Imm(0),
/* |VR| /* void* PAGE_LIMIT _Imm(0),
/* |VR| /* void* SMALLEST_BLOCK_SIZE _Imm(0),
/* |VR| %C_S| /* DXX& ZONE_NAME_RefZero(),
/* |PR| PROC|BO|AS| %IMM| /* uint32_t GET_PAGE = 0,
/* |PR| PROC|BO|AS| %IMM| /* uint32_t FREE_PAGE = 0) ;
```

```
extern "C" void OTS$MOVE3(
/* |PR| %IMM| /* int32_t LEN ,
/* |VR| US| %REF| /* DXX& SRC_STR ,
/* |VR| US| %REF| /* DXX& DST_STR ) ;
```

```
extern "C" int32_t LIB$ADDX(
/* |VR| US| %REF| /* void* ADDEND_ARRAY ,
/* |VR| US| %REF| /* void* AUGEND_ARRAY ,
/* |VR| US| VL| %REF| /* void* RESULTANT_ARRAY ,
/* |VR| /* int32_t* ARRAY_LENGTH _Imm(0)) ;
```

```
extern "C" int32_t LIB$ADD_TIMES(
/* |VR| /* void* TIME1 ,
/* |VR| /* void* TIME2 ,
/* |VR| VL| /* void* RESULTANT_TIME ) ;
```

```

%REF augend_array : [UNSAFE] ARRAY [$l2..$u2:INTEGER] OF
$UBYTE;
%REF resultant_array : [VOLATILE,UNSAFE] ARRAY
[$l3..$u3:INTEGER] OF $UBYTE;
    array_length : INTEGER := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$add_times (
    time1 : $UQUAD;
    time2 : $UQUAD;
    VAR resultant_time : [VOLATILE] $UQUAD) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$analyze_sdesc (
    input_descriptor : $UQUAD;
    VAR data_length : [VOLATILE] $UWORD;
    VAR data_address : [VOLATILE] $DEFPTR) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$analyze_sdesc_64 (
    input_descriptor : $UQUAD;
    VAR data_length : [VOLATILE] $UQUAD;
    VAR data_address : [VOLATILE] lib$routines$$typl;
    VAR descriptor_type : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$ascii_to_uid (
    input_string : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER]
OF CHAR;
    VAR output_uid : [VOLATILE] $UOCTA) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$asn_wth_mbx (
    device_name : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER]
OF CHAR;
    maximum_message_size : INTEGER := %IMMED 0;
    buffer_quota : INTEGER := %IMMED 0;
    VAR device_channel : [VOLATILE] $UWORD;
    VAR mailbox_channel : [VOLATILE] $UWORD) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$ast_in_prog : BOOLEAN; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$attach (
    process_id : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$bbcci (
    position : INTEGER;

```

```

extern "C" int32_t LIB$ANALYZE_SDESC(
/* |VR| */ void* INPUT_DESCRIPTOR ,
/* |VR|VL| */ uint16_t* DATA_LENGTH ,
/* |VR|VL| */ $DEFTYP* DATA_ADDRESS ) ;

extern "C" int32_t LIB$ANALYZE_SDESC_64(
/* |VR| */ void* INPUT_DESCRIPTOR ,
/* |VR|VL| */ void* DATA_LENGTH ,
/* |VR|VL| */ void* DATA_ADDRESS ,
/* |VR|VL| */ uint16_t* DESCRIPTOR_TYPE _Imm(0))
;

extern "C" int32_t LIB$ASCII_TO_UID(
/* |VR|&C_S| */ DXX& INPUT_STRING ,
/* |VR|VL| */ void* OUTPUT_UID ) ;

extern "C" int32_t LIB$ASN_WTH_MBX(
/* |VR|&C_S| */ DXX& DEVICE_NAME ,
/* |VR| */ int32_t* MAXIMUM_MESSAGE_SIZE
_Imm(0),
/* |VR| */ int32_t* BUFFER_QUOTA _Imm(0),
/* |VR|VL| */ uint16_t* DEVICE_CHANNEL _Imm(0),
/* |VR|VL| */ uint16_t* MAILBOX_CHANNEL _Imm(0))
;

extern "C" bool LIB$AST_IN_PROG(void) ;

extern "C" int32_t LIB$ATTACH(
/* |VR| */ uint32_t* PROCESS_ID ) ;

extern "C" int32_t LIB$BBCCI(
/* |VR| */ int32_t* POSITION ,
/* |PR|&IMM| */ $DEFTYP BIT_ZERO_ADDRESS ) ;

extern "C" int32_t LIB$BBSSI(
/* |VR| */ int32_t* POSITION ,
/* |PR|&IMM| */ $DEFTYP BIT_ZERO_ADDRESS ) ;

extern "C" int32_t LIB$BUILD_NODESPEC(
/* |VR|&C_S| */ DXX& PRIMARY_NODENAME ,
/* |VR|VL|&C_S| */ DXX& NODESPEC ,
/* |VR|&C_S| */ DXX& ACS_RefZero(),

```

```

%IMMED bit_zero_address : $DEFPTR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$bbssi (
    position : INTEGER;
    %IMMED bit_zero_address : $DEFPTR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$build_nodspec (
    primary_nodename : [CLASS_S] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
    VAR nodspec : [CLASS_S,VOLATILE] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR;
    acs : [CLASS_S] PACKED ARRAY [$13..$u3:INTEGER] OF CHAR
:= %IMMED 0;
    secondary_nodename : [CLASS_S] PACKED ARRAY
[$14..$u4:INTEGER] OF CHAR := %IMMED 0;
    VAR nodspec_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$callg (
    %REF argument_list : [UNSAFE] ARRAY [$11..$u1:INTEGER] OF
$UBYTE;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE user_procedure)
: INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$callg_64 (
    %REF argument_list : ARRAY [$11..$u1:INTEGER] OF $UQUAD;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE user_procedure)
: INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$char (
    VAR one_character_string : [CLASS_S,VOLATILE] PACKED
ARRAY [$11..$u1:INTEGER] OF CHAR;
    ascii_code : $UBYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$compare_nodename (
    nodename1 : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
CHAR;
    nodename2 : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER] OF
CHAR;
    VAR comparison_result : [VOLATILE] UNSIGNED) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$compare_uid (
    first_uid : $UOCTA;
    second_uid : $UOCTA;

```

```

/* |VR| %C_S|          */ DXX& SECONDARY_NODENAME _RefZero(),
/* |VR| VL|           */ uint16_t* NODESPEC_LENGTH _Imm(0)
;

extern "C" int32_t LIB$CALLG(
/* |VR| US| %REF|     */ void* ARGUMENT_LIST ,
/* |PR| PROC|BO|AS| %IMM| */ uint32_t USER_PROCEDURE ) ;

extern "C" int32_t LIB$CALLG_64(
/* |VR| %REF|        */ void* ARGUMENT_LIST ,
/* |PR| PROC|BO|AS| %IMM| */ uint32_t USER_PROCEDURE ) ;

extern "C" int32_t LIB$CHAR(
/* |VR| VL| %C_S|    */ DXX& ONE_CHARACTER_STRING ,
/* |VR|              */ uchar_t* ASCII_CODE ) ;

extern "C" int32_t LIB$COMPARE_NODENAME(
/* |VR| %C_S|        */ DXX& NODENAME1 ,
/* |VR| %C_S|        */ DXX& NODENAME2 ,
/* |VR| VL|         */ uint32_t* COMPARISON_RESULT ) ;

extern "C" int32_t LIB$COMPARE_UID(
/* |VR|              */ void* FIRST_UID ,
/* |VR|              */ void* SECOND_UID ,
/* |VR| VL|         */ int32_t* MATCH_RESULT ) ;

extern "C" int32_t LIB$COMPRESS_NODENAME(
/* |VR| %C_S|        */ DXX& NODENAME ,
/* |VR| VL| %C_S|    */ DXX& COMPRESSED_NODENAME ,
/* |VR| VL|         */ uint16_t* RESULTANT_LENGTH _Imm(0)
;

extern "C" int32_t LIB$CONVERT_DATE_STRING(
/* |VR| %C_S|        */ DXX& DATE_STRING ,
/* |VR| VL|         */ void* DATE_TIME ,
/* |VR| VL|         */ uint32_t* USER_CONTEXT _Imm(0),
/* |VR|           */ uint32_t* FLAGS _Imm(0),
/* |VR| %REF|       */ void* DEFAULTS _Imm(0),
/* |VR| VL|         */ uint32_t* DEFAULTTED_FIELDS _Imm(0)
;

extern "C" int32_t LIB$CRC(
/* |VR| %REF|        */ void* CRC_TABLE ,
/* |VR|              */ int32_t* INITIAL_CRC ,

```

```

VAR match_result : [VOLATILE] INTEGER) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$compress_nodename (
    nodename : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
    CHAR;
    VAR compressed_nodename : [CLASS_S,VOLATILE] PACKED ARRAY
    [$12..$u2:INTEGER] OF CHAR;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
    INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$convert_date_string (
    date_string : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
    OF CHAR;
    VAR date_time : [VOLATILE] $UQUAD;
    VAR user_context : [VOLATILE] UNSIGNED := %IMMED 0;
    flags : UNSIGNED := %IMMED 0;
    %REF defaults : ARRAY [$15..$u5:INTEGER] OF $UWORD :=
    %IMMED 0;
    VAR defaulted_fields : [VOLATILE] UNSIGNED := %IMMED 0) :
    INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$crc (
    %REF crc_table : ARRAY [$11..$u1:INTEGER] OF INTEGER;
    initial_crc : INTEGER;
    stream : [CLASS_S] PACKED ARRAY [$13..$u3:INTEGER] OF
    CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] PROCEDURE lib$crc_table (
    polynomial_coefficient : UNSIGNED;
    %REF crc_table : [VOLATILE] ARRAY [$12..$u2:INTEGER] OF
    INTEGER); EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$create_dir (
    device_directory_spec : [CLASS_S] PACKED ARRAY
    [$11..$u1:INTEGER] OF CHAR;
    owner_UIC : UNSIGNED := %IMMED 0;
    protection_enable : $UWORD := %IMMED 0;
    protection_value : $UWORD := %IMMED 0;
    maximum_versions : $UWORD := %IMMED 0;
    relative_volume_number : $UWORD := %IMMED 0) : INTEGER;
    EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$create_user_vm_zone (
    VAR zone_id : [VOLATILE] UNSIGNED;
    user_argument : UNSIGNED := %IMMED 0;

```

```

/* |VR|%C_S| */ DXX& STREAM ) ;

extern "C" void LIB$CRC_TABLE(
/* |VR| */ /* uint32_t* POLYNOMIAL_COEFFICIENT ,
/* |VR|VL|%REF| */ void* CRC_TABLE ) ;

extern "C" int32_t LIB$CREATE_DIR(
/* |VR|%C_S| */ /* DXX& DEVICE_DIRECTORY_SPEC ,
/* |VR| */ /* uint32_t* OWNER_UIC _Imm(0),
/* |VR| */ /* uint16_t* PROTECTION_ENABLE
 Imm(0),
/* |VR| */ /* uint16_t* PROTECTION_VALUE _Imm(0),
/* |VR| */ /* uint16_t* MAXIMUM_VERSIONS _Imm(0),
/* |VR| */ /* uint16_t* RELATIVE_VOLUME_NUMBER
 Imm(0) ) ;

extern "C" int32_t LIB$CREATE_USER_VM_ZONE(
/* |VR|VL| */ /* uint32_t* ZONE_ID ,
/* |VR| */ /* uint32_t* USER_ARGUMENT _Imm(0),
/* |PR|PROC|BO|AS|%IMM| */ /* uint32_t USER_ALLOCATION_PROCEDURE
= 0,
/* |PR|PROC|BO|AS|%IMM| */ /* uint32_t
USER_DEALLOCATION_PROCEDURE = 0,
/* |PR|PROC|BO|AS|%IMM| */ /* uint32_t USER_RESET_PROCEDURE = 0,
/* |PR|PROC|BO|AS|%IMM| */ /* uint32_t USER_DELETE_PROCEDURE =
0,
/* |VR|%C_S| */ /* DXX& ZONE_NAME _RefZero() ) ;

extern "C" int32_t LIB$CREATE_USER_VM_ZONE_64(
/* |VR|VL| */ /* void* ZONE_ID ,
/* |VR| */ /* void* USER_ARGUMENT _Imm(0),
/* |PR|PROC|BO|AS|%IMM| */ /* uint32_t USER_ALLOCATION_PROCEDURE
= 0,
/* |PR|PROC|BO|AS|%IMM| */ /* uint32_t
USER_DEALLOCATION_PROCEDURE = 0,
/* |PR|PROC|BO|AS|%IMM| */ /* uint32_t USER_RESET_PROCEDURE = 0,
/* |PR|PROC|BO|AS|%IMM| */ /* uint32_t USER_DELETE_PROCEDURE =
0,
/* |VR|%C_S| */ /* DXX& ZONE_NAME _RefZero() ) ;

extern "C" int32_t LIB$CREATE_VM_ZONE(
/* |VR|VL| */ /* uint32_t* ZONE_ID ,
/* |VR| */ /* int32_t* ALGORITHM _Imm(0),
/* |VR| */ /* int32_t* ALGORITHM_ARGUMENT
 Imm(0),
/* |VR| */ /* uint32_t* FLAGS _Imm(0),
/* |VR| */ /* int32_t* EXTEND_SIZE _Imm(0),

```

```

    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_allocation_procedure := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_deallocation_procedure := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_reset_procedure := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_delete_procedure := %IMMED 0;
    zone_name : [CLASS_S] PACKED ARRAY [$17..$u7:INTEGER] OF
CHAR := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$create_user_vm_zone_64 (
    VAR zone_id : [VOLATILE] $UQUAD;
    user_argument : $UQUAD := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_allocation_procedure := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_deallocation_procedure := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_reset_procedure := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_delete_procedure := %IMMED 0;
    zone_name : [CLASS_S] PACKED ARRAY [$17..$u7:INTEGER] OF
CHAR := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$create_vm_zone (
    VAR zone_id : [VOLATILE] UNSIGNED;
    algorithm : INTEGER := %IMMED 0;
    algorithm_argument : INTEGER := %IMMED 0;
    flags : UNSIGNED := %IMMED 0;
    extend_size : INTEGER := %IMMED 0;
    initial_size : INTEGER := %IMMED 0;
    block_size : INTEGER := %IMMED 0;
    alignment : INTEGER := %IMMED 0;
    page_limit : INTEGER := %IMMED 0;
    smallest_block_size : INTEGER := %IMMED 0;
    zone_name : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
OF CHAR := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE get_page :=
%IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE free_page :=
%IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] PROCEDURE lib$crf_ins_key (
    %REF control_table : ARRAY [$11..$u1:INTEGER] OF INTEGER;

```

```

/* |VR|          */ int32_t* INITIAL_SIZE _Imm(0),
/* |VR|          */ int32_t* BLOCK_SIZE _Imm(0),
/* |VR|          */ int32_t* ALIGNMENT _Imm(0),
/* |VR|          */ int32_t* PAGE_LIMIT _Imm(0),
/* |VR|          */ int32_t* SMALLEST_BLOCK_SIZE
_Imm(0),
/* |VR| %C_S|    */ DXX& ZONE_NAME_RefZero(),
/* |PR|PROC|BO|AS| %IMM| */ uint32_t GET_PAGE = 0,
/* |PR|PROC|BO|AS| %IMM| */ uint32_t FREE_PAGE = 0) ;

extern "C" void LIB$CRF_INS_KEY(
/* |VR| %REF|    */ void* CONTROL_TABLE ,
/* |VR| %C_S|    */ DXX& KEY_STRING ,
/* |VR|          */ int32_t* SYMBOL_VALUE ,
/* |VR|          */ uint32_t* FLAGS ) ;

extern "C" void LIB$CRF_INS_REF(
/* |VR| %REF|    */ void* CONTROL_TABLE ,
/* |VR|          */ int32_t* LONGWORD_INTEGER_KEY ,
/* |VR| %C_S|    */ DXX& REFERENCE_STRING ,
/* |VR| VL|      */ int32_t* LONGWORD_INTEGER_REFERENCE
,
/* |VR|          */ int32_t* REF_DEFINITION_INDICATOR )
;

extern "C" void LIB$CRF_OUTPUT(
/* |VR| %REF|    */ void* CONTROL_TABLE ,
/* |VR|          */ int32_t* OUTPUT_LINE_WIDTH ,
/* |VR|          */ int32_t* PAGE1 ,
/* |VR|          */ int32_t* PAGE2 ,
/* |VR|          */ int32_t* MODE_INDICATOR ,
/* |VR|          */ int32_t* DELETE_SAVE_INDICATOR ) ;

extern "C" int32_t LIB$CURRENCY(
/* |VR| VL| %C_S| */ DXX& CURRENCY_STRING ,
/* |VR| VL|          */ uint16_t* RESULTANT_LENGTH _Imm(0))
;

extern "C" int32_t LIB$CVTF_FROM_INTERNAL_TIME(
/* |VR|          */ uint32_t* OPERATION ,
/* |VR| VL|        */ float* RESULTANT_TIME ,
/* |VR|          */ void* INPUT_TIME ) ;

extern "C" int32_t LIB$CVTF_TO_INTERNAL_TIME(
/* |VR|          */ uint32_t* OPERATION ,
/* |VR|          */ float* INPUT_TIME ,

```

```

key_string : [CLASS_S] PACKED ARRAY [$l2..$u2:INTEGER] OF
CHAR;
symbol_value : INTEGER;
flags : UNSIGNED); EXTERNAL;

[ASYNCHRONOUS] PROCEDURE lib$crf_ins_ref (
%REF control_table : ARRAY [$l1..$u1:INTEGER] OF INTEGER;
longword_integer_key : INTEGER;
reference_string : [CLASS_S] PACKED ARRAY
[$l3..$u3:INTEGER] OF CHAR;
VAR longword_integer_reference : [VOLATILE] INTEGER;
ref_definition_indicator : INTEGER); EXTERNAL;

[ASYNCHRONOUS] PROCEDURE lib$crf_output (
%REF control_table : ARRAY [$l1..$u1:INTEGER] OF INTEGER;
output_line_width : INTEGER;
page1 : INTEGER;
page2 : INTEGER;
mode_indicator : INTEGER;
delete_save_indicator : INTEGER); EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$currency (
VAR currency_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$l1..$u1:INTEGER] OF CHAR;
VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvtf_from_internal_time (
operation : UNSIGNED;
VAR resultant_time : [VOLATILE] SINGLE;
input_time : $UQUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvtf_to_internal_time (
operation : UNSIGNED;
input_time : SINGLE;
VAR resultant_time : [VOLATILE] $UQUAD) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvtf_dtb (
%IMMED byte_count : INTEGER;
%REF numeric_string : PACKED ARRAY [$l2..$u2:INTEGER] OF
CHAR;
VAR result : [VOLATILE] INTEGER) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvtf_dtb_64 (

```

```

/* |VR|VL| */ void* RESULTANT_TIME ) ;

extern "C" int32_t LIB$CVT_DTB(
/* |PR|%IMM| */ int32_t BYTE_COUNT ,
/* |VR|%REF| */ DXX& NUMERIC_STRING ,
/* |VR|VL| */ int32_t* RESULT ) ;

extern "C" int32_t LIB$CVT_DTB_64(
/* |PR|%IMM| */ int32_t BYTE_COUNT ,
/* |VR|%REF| */ DXX& NUMERIC_STRING ,
/* |VR|VL| */ void* RESULT ) ;

extern "C" int32_t LIB$CVT_DX_DX(
/* |VR|US|%REF| */ void* SOURCE_ITEM ,
/* |VR|US|VL|%REF| */ void* DESTINATION_ITEM ,
/* |VR|VL| */ uint16_t* WORD_INTEGER_DEST_LENGTH
_Imm(0) ) ;

extern "C" int32_t LIB$CVT_FROM_INTERNAL_TIME(
/* |VR| */ uint32_t* OPERATION ,
/* |VR|VL| */ uint32_t* RESULTANT_TIME ,
/* |VR| */ void* INPUT_TIME _Imm(0) ) ;

extern "C" int32_t LIB$CVT_HTB(
/* |PR|%IMM| */ int32_t BYTE_COUNT ,
/* |VR|%REF| */ DXX& NUMERIC_STRING ,
/* |VR|VL| */ int32_t* RESULT ) ;

extern "C" int32_t LIB$CVT_HTB_64(
/* |PR|%IMM| */ int32_t BYTE_COUNT ,
/* |VR|%REF| */ DXX& NUMERIC_STRING ,
/* |VR|VL| */ void* RESULT ) ;

extern "C" int32_t LIB$CVT_OTB(
/* |PR|%IMM| */ int32_t BYTE_COUNT ,
/* |VR|%REF| */ DXX& NUMERIC_STRING ,
/* |VR|VL| */ int32_t* RESULT ) ;

extern "C" int32_t LIB$CVT_OTB_64(
/* |PR|%IMM| */ int32_t BYTE_COUNT ,
/* |VR|%REF| */ DXX& NUMERIC_STRING ,
/* |VR|VL| */ void* RESULT ) ;

```



```

    %IMMED byte_count : INTEGER;
    %REF numeric_string : PACKED ARRAY [$l2..$u2:INTEGER] OF
CHAR;
    VAR result : [VOLATILE] $QUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvdt_dx_dx (
    %REF source_item : [UNSAFE] ARRAY [$l1..$u1:INTEGER] OF
$SUBYTE;
    %REF destination_item : [VOLATILE,UNSAFE] ARRAY
[$l2..$u2:INTEGER] OF $SUBYTE;
    VAR word_integer_dest_length : [VOLATILE] $UWORD :=
%IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvdt_from_internal_time (
    operation : UNSIGNED;
    VAR resultant_time : [VOLATILE] UNSIGNED;
    input_time : $QUAD := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvdt_htb (
    %IMMED byte_count : INTEGER;
    %REF numeric_string : PACKED ARRAY [$l2..$u2:INTEGER] OF
CHAR;
    VAR result : [VOLATILE] INTEGER) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvdt_htb_64 (
    %IMMED byte_count : INTEGER;
    %REF numeric_string : PACKED ARRAY [$l2..$u2:INTEGER] OF
CHAR;
    VAR result : [VOLATILE] $QUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvdt_otb (
    %IMMED byte_count : INTEGER;
    %REF numeric_string : PACKED ARRAY [$l2..$u2:INTEGER] OF
CHAR;
    VAR result : [VOLATILE] INTEGER) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvdt_otb_64 (
    %IMMED byte_count : INTEGER;
    %REF numeric_string : PACKED ARRAY [$l2..$u2:INTEGER] OF
CHAR;
    VAR result : [VOLATILE] $QUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$cvdt_to_internal_time (
    operation : UNSIGNED;
    input_time : INTEGER;

```

```

extern "C" int32_t LIB$CVDT_TO_INTERNAL_TIME(
/* |VR|                               */ uint32_t* OPERATION ,
/* |VR|                               */ int32_t* INPUT_TIME ,
/* |VR|VL|                             */ void* RESULTANT_TIME ) ;

extern "C" int32_t LIB$CVDT_VECTIM(
/* |VR|%REF|                           */ void* INPUT_TIME ,
/* |VR|VL|                             */ void* RESULTANT_TIME ) ;

extern "C" int32_t LIB$DATE_TIME(
/* |VR|VL|%C_S|                         */ DXX& DATE_TIME_STRING ) ;

extern "C" int32_t LIB$DAY(
/* |VR|VL|                             */ int32_t* NUMBER_OF_DAYS ,
/* |VR|                               */ void* USER_TIME _Imm(0),
/* |VR|VL|                             */ int32_t* DAY_TIME _Imm(0)) ;

extern "C" int32_t LIB$DAY_OF_WEEK(
/* |VR|                               */ void* USER_TIME _Imm(0),
/* |VR|VL|                             */ uint32_t* DAY_NUMBER _Imm(0)) ;

extern "C" int32_t LIB$DECODE_FAULT(
/* |VR|US|%REF|                         */ void* SIGNAL_ARGUMENTS ,
/* |VR|US|%REF|                         */ void* MECHANISM_ARGUMENTS ,
/* |PR|PROC|BO|AS|%IMM|                 */ uint32_t USER_PROCEDURE ,
/* |PR|US|%IMM|                         */ int32_t UNSPECIFIED_USER_ARGUMENT
= 0,
/* |VR|%REF|                             */ void* INSTRUCTION_DEFINITIONS
_Imm(0)) ;

extern "C" int32_t LIB$DEC_OVER(
/* |VR|                               */ uint32_t* NEW_SETTING ) ;

extern "C" int32_t LIB$DELETE_FILE(
/* |VR|%C_S|                             */ DXX& FILESPEC ,
/* |VR|%C_S|                             */ DXX& DEFAULT_FILESPEC_RefZero(),
/* |VR|%C_S|                             */ DXX& RELATED_FILESPEC_RefZero(),
/* |PR|PROC|BO|AS|%IMM|                 */ uint32_t USER_SUCCESS_PROCEDURE =
0,
/* |PR|PROC|BO|AS|%IMM|                 */ uint32_t USER_ERROR_PROCEDURE = 0,
/* |PR|PROC|BO|AS|%IMM|                 */ uint32_t USER_CONFIRM_PROCEDURE =
0,
/* |PR|US|%IMM|                         */ int32_t USER_SPECIFIED_ARGUMENT =
0,
/* |VR|VL|%C_S|                         */ DXX& RESULTANT_NAME_RefZero(),

```

```

VAR resultant_time : [VOLATILE] $UQUAD) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$svt_vectim (
    %REF input_time : ARRAY [$l1..$u1:INTEGER] OF $UWORD;
    VAR resultant_time : [VOLATILE] $UQUAD) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$date_time (
    VAR date_time_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$l1..$u1:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$day (
    VAR number_of_days : [VOLATILE] INTEGER;
    user_time : $UQUAD := %IMMED 0;
    VAR day_time : [VOLATILE] INTEGER := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$day_of_week (
    user_time : $UQUAD := %IMMED 0;
    VAR day_number : [VOLATILE] UNSIGNED) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$decode_fault (
    %REF signal_arguments : [UNSAFE] ARRAY [$l1..$u1:INTEGER]
OF $SUBYTE;
    %REF mechanism_arguments : [UNSAFE] ARRAY
[$l2..$u2:INTEGER] OF $SUBYTE;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE user_procedure;
    %IMMED unspecified_user_argument : [UNSAFE] INTEGER :=
%IMMED 0;
    %REF instruction_definitions : ARRAY [$l5..$u5:INTEGER]
OF $SUBYTE := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$dec_over (
    new_setting : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$delete_file (
    filespec : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER] OF
CHAR;
    default_filespec : [CLASS_S] PACKED ARRAY
[$l2..$u2:INTEGER] OF CHAR := %IMMED 0;
    related_filespec : [CLASS_S] PACKED ARRAY
[$l3..$u3:INTEGER] OF CHAR := %IMMED 0;

```

```

/* |VR|VL|                               */ uint32_t* FILE_SCAN_CONTEXT
 Imm(0),
/* |VR|                                   */ uint32_t* FLAGS_Imm(0) );

extern "C" int32_t LIB$DELETE_LOGICAL(
/* |VR| %C_S|                             */ DXX& LOGICAL_NAME ,
/* |VR| %C_S|                             */ DXX& TABLE_NAME_RefZero() );

extern "C" int32_t LIB$DELETE_SYMBOL(
/* |VR| %C_S|                             */ DXX& SYMBOL ,
/* |VR|                                   */ int32_t* TABLE_TYPE_INDICATOR
 Imm(0) );

extern "C" int32_t LIB$DELETE_VM_ZONE(
/* |VR|                                   */ uint32_t* ZONE_ID );

extern "C" int32_t LIB$DELETE_VM_ZONE_64(
/* |VR|                                   */ void* ZONE_ID );

extern "C" int32_t LIB$DIGIT_SEP(
/* |VR|VL| %C_S|                         */ DXX& DIGIT_SEPARATOR_STRING ,
/* |VR|VL|                               */ uint16_t* RESULTANT_LENGTH_Imm(0) );

extern "C" int32_t LIB$DISABLE_CTRL(
/* |VR|                                   */ uint32_t* DISABLE_MASK ,
/* |VR|VL|                               */ uint32_t* OLD_MASK_Imm(0) );

extern "C" int32_t LIB$DO_COMMAND(
/* |VR| %C_S|                             */ DXX& COMMAND_STRING );

extern "C" int32_t LIB$EDIV(
/* |VR|                                   */ int32_t* LONGWORD_INTEGER_DIVISOR ,
/* |VR|                                   */ void* QUADWORD_INTEGER_DIVIDEND ,
/* |VR|VL|                               */ int32_t* LONGWORD_INTEGER_QUOTIENT
,
/* |VR|VL|                               */ int32_t* REMAINDER );

extern "C" int32_t LIB$EMODD(
/* |VR|                                   */ double* FLOATING_POINT_MULTIPLIER ,
/* |VR|                                   */ uchar_t* MULTIPLIER_EXTENSION ,
/* |VR|                                   */ double* FLOATING_POINT_MULTIPLICAND
,

```

```

%IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_success_procedure := %IMMED 0;
%IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_error_procedure := %IMMED 0;
%IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_confirm_procedure := %IMMED 0;
%IMMED user_specified_argument : [UNSAFE] INTEGER :=
%IMMED 0;
VAR resultant_name : [CLASS_S,VOLATILE] PACKED ARRAY
[$18..$u8:INTEGER] OF CHAR := %IMMED 0;
VAR file_scan_context : [VOLATILE] UNSIGNED := %IMMED 0;
flags : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$delete_logical (
logical_name : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
OF CHAR;
table_name : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER] OF
CHAR := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$delete_symbol (
symbol : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
CHAR;
table_type_indicator : INTEGER := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$delete_vm_zone (
zone_id : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$delete_vm_zone_64 (
zone_id : $UQUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$digit_sep (
VAR digit_separator_string : [CLASS_S,VOLATILE] PACKED
ARRAY [$11..$u1:INTEGER] OF CHAR;
VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$disable_ctrl (
disable_mask : UNSIGNED;
VAR old_mask : [VOLATILE] UNSIGNED := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$do_command (
command_string : [CLASS_S] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

```

```

/* |VR|VL| */ int32_t* INTEGER_PORTION ,
/* |VR|VL| */ double* FRACTIONAL_PORTION ) ;

extern "C" int32_t LIB$EMODF(
/* |VR| */ /* float* FLOATING_POINT_MULTIPLIER ,
/* |VR| */ /* uchar_t* MULTIPLIER_EXTENSION ,
/* |VR| */ /* float* FLOATING_POINT_MULTIPLICAND
,
/* |VR|VL| */ int32_t* INTEGER_PORTION ,
/* |VR|VL| */ float* FRACTIONAL_PORTION ) ;

extern "C" int32_t LIB$EMODG(
/* |VR| */ /* double* FLOATING_POINT_MULTIPLIER ,
/* |VR| */ /* uint16_t* MULTIPLIER_EXTENSION ,
/* |VR| */ /* double* FLOATING_POINT_MULTIPLICAND
,
/* |VR|VL| */ int32_t* INTEGER_PORTION ,
/* |VR|VL| */ double* FRACTIONAL_PORTION ) ;

extern "C" int32_t LIB$EMODH(
/* |VR| */ /* int64_t* FLOATING_POINT_MULTIPLIER
,
/* |VR| */ /* uint16_t* MULTIPLIER_EXTENSION ,
/* |VR| */ /* int64_t*
FLOATING_POINT_MULTIPLICAND ,
/* |VR|VL| */ int32_t* INTEGER_PORTION ,
/* |VR|VL| */ int64_t* FRACTIONAL_PORTION ) ;

extern "C" int32_t LIB$EMUL(
/* |VR| */ /* int32_t*
LONGWORD_INTEGER_MULTIPLIER ,
/* |VR| */ /* int32_t*
LONGWORD_INTEGER_MULTIPLICAND ,
/* |VR| */ /* int32_t* ADDEND ,
/* |VR|VL| */ /* void* PRODUCT ) ;

extern "C" int32_t LIB$ENABLE_CTRL(
/* |VR| */ /* uint32_t* ENABLE_MASK ,
/* |VR|VL| */ /* uint32_t* OLD_MASK _Imm(0) ) ;

extern "C" int32_t LIB$EXPAND_NODENAME(
/* |VR|&C_S| */ /* DXX& NODENAME ,
/* |VR|VL|&C_S| */ /* DXX& FULLNAME ,
/* |VR|VL| */ /* uint16_t* RESULTANT_LENGTH _Imm(0)
;

```

```
[ASYNCHRONOUS] FUNCTION lib$ediv (
    longword_integer_divisor : INTEGER;
    quadword_integer_dividend : $QUAD;
    VAR longword_integer_quotient : [VOLATILE] INTEGER;
    VAR remainder : [VOLATILE] INTEGER) : INTEGER; EXTERNAL;
```

```
[ASYNCHRONOUS] FUNCTION lib$emodd (
    floating_point_multiplier : D_FLOAT$$TYPE;
    multiplier_extension : $BYTE;
    floating_point_multiplicand : D_FLOAT$$TYPE;
    VAR integer_portion : [VOLATILE] INTEGER;
    VAR fractional_portion : [VOLATILE] D_FLOAT$$TYPE) :
INTEGER; EXTERNAL;
```

```
[ASYNCHRONOUS] FUNCTION lib$emodf (

    floating_point_multiplier : SINGLE;
    multiplier_extension : $BYTE;
    floating_point_multiplicand : SINGLE;
    VAR integer_portion : [VOLATILE] INTEGER;
    VAR fractional_portion : [VOLATILE] SINGLE) : INTEGER;
EXTERNAL;
```

```
[ASYNCHRONOUS] FUNCTION lib$emodg (
    floating_point_multiplier : G_FLOAT$$TYPE;
    multiplier_extension : $WORD;
    floating_point_multiplicand : G_FLOAT$$TYPE;
    VAR integer_portion : [VOLATILE] INTEGER;
    VAR fractional_portion : [VOLATILE] G_FLOAT$$TYPE) :
INTEGER; EXTERNAL;
```

```
[ASYNCHRONOUS] FUNCTION lib$emodh (
    floating_point_multiplier : QUADRUPLE;
    multiplier_extension : $WORD;
    floating_point_multiplicand : QUADRUPLE;
    VAR integer_portion : [VOLATILE] INTEGER;
    VAR fractional_portion : [VOLATILE] QUADRUPLE) : INTEGER;
EXTERNAL;
```

```
[ASYNCHRONOUS] FUNCTION lib$emul (
    longword_integer_multiplier : INTEGER;
    longword_integer_multiplicand : INTEGER;
    addend : INTEGER;
    VAR product : [VOLATILE] $QUAD) : INTEGER; EXTERNAL;
```

```
extern "C" int32_t LIB$EXTV(
/* |VR|          */ int32_t* POSITION ,
/* |VR|          */ uchar_t* SIZE ,
/* |PR| %IMM|    */ $DEFTYP BASE_ADDRESS ) ;
```

```
extern "C" int32_t LIB$EXTZV(
/* |VR|          */ int32_t* POSITION ,
/* |VR|          */ uchar_t* SIZE ,
/* |PR| %IMM|    */ $DEFTYP BASE_ADDRESS ) ;
```

```
extern "C" int32_t LIB$FFC(
/* |VR|          */ int32_t* POSITION ,
/* |VR|          */ uchar_t* SIZE ,
/* |VR|          */ $DEFTYP* BASE ,
/* |VR| VL|      */ int32_t* FIND_POSITION ) ;
```

```
extern "C" int32_t LIB$FFS(
/* |VR|          */ int32_t* POSITION ,
/* |VR|          */ uchar_t* SIZE ,
/* |VR|          */ $DEFTYP* BASE ,
/* |VR| VL|      */ int32_t* FIND_POSITION ) ;
```

```
extern "C" int32_t LIB$FID_TO_NAME(
/* |VR| %C_S|    */ DXX& DEVICE_NAME ,
/* |VR| %REF|    */ void* FILE_ID ,
/* |VR| VL| %C_S| */ DXX& FILESPEC ,
/* |VR| VL|      */ uint16_t* FILESPEC_LENGTH _Imm(0),
/* |VR| %REF|    */ void* DIRECTORY_ID _Imm(0),
/* |VR| VL|      */ uint32_t* ACP_STATUS _Imm(0)) ;
```

```
extern "C" int32_t LIB$FILE_SCAN(
/* |VR| US| %REF| */ void* FAB ,
/* |PR| PROC|BO|AS| %IMM| */ uint32_t USER_SUCCESS_PROCEDURE ,
/* |PR| PROC|BO|AS| %IMM| */ uint32_t USER_ERROR_PROCEDURE ,
/* |VR| VL|      */ uint32_t* CONTEXT _Imm(0)) ;
```

```
extern "C" int32_t LIB$FILE_SCAN_END(
/* |VR| US| VL| %REF| */ void* FAB _Imm(0),
/* |VR| VL|          */ uint32_t* CONTEXT _Imm(0)) ;
```

```
extern "C" int32_t LIB$FIND_FILE(
/* |VR| %C_S|    */ DXX& FILESPEC ,
/* |VR| VL| %C_S| */ DXX& RESULTANT_FILESPEC ,
/* |VR| VL|      */ uint32_t* CONTEXT ,
```

```

[ASYNCHRONOUS] FUNCTION lib$enable_ctrl (
    enable_mask : UNSIGNED;
    VAR old_mask : [VOLATILE] UNSIGNED := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$expand_nodename (
    nodename : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER] OF
CHAR;
    VAR fullname : [CLASS_S,VOLATILE] PACKED ARRAY
[$l2..$u2:INTEGER] OF CHAR;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$extv (
    position : INTEGER;
    size : $BYTE;
    %IMMED base_address : $DEFPTR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$extzv (
    position : INTEGER;
    size : $BYTE;
    %IMMED base_address : $DEFPTR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$ffc (
    position : INTEGER;
    size : $BYTE;
    base : $DEFPTR;
    VAR find_position : [VOLATILE] INTEGER) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$ffs (
    position : INTEGER;
    size : $BYTE;
    base : $DEFPTR;
    VAR find_position : [VOLATILE] INTEGER) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$fid_to_name (
    device_name : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER]
OF CHAR;
    %REF file_id : ARRAY [$l2..$u2:INTEGER] OF $UWORD;
    VAR filespec : [CLASS_S,VOLATILE] PACKED ARRAY
[$l3..$u3:INTEGER] OF CHAR;
    VAR filespec_length : [VOLATILE] $UWORD := %IMMED 0;

```

```

/* |VR| %C_S|          */ DXX& DEFAULT_FILESPEC _RefZero(),
/* |VR| %C_S|          */ DXX& RELATED_FILESPEC _RefZero(),
/* |VR| VL|           */ uint32_t* STATUS_VALUE _Imm(0),
/* |VR|           */ uint32_t* FLAGS _Imm(0) );

extern "C" int32_t LIB$FIND_FILE_END(
/* |VR|           */ uint32_t* CONTEXT );

extern "C" int32_t LIB$FIND_IMAGE_SYMBOL(
/* |VR| %C_S|          */ DXX& FILENAME ,
/* |VR| %C_S|          */ DXX& SYMBOL ,
/* |VR| VL|           */ int32_t* SYMBOL_VALUE ,
/* |VR| %C_S|          */ DXX& IMAGE_NAME _RefZero(),
/* |PR| %IMM|         */ uint32_t FLAGS = 0) ;

extern "C" int32_t LIB$FIND_VM_ZONE(
/* |VR| VL|           */ uint32_t* CONTEXT ,
/* |VR| VL|           */ uint32_t* ZONE_ID );

extern "C" int32_t LIB$FIND_VM_ZONE_64(
/* |VR| VL|           */ void* CONTEXT ,
/* |VR| VL|           */ void* ZONE_ID );

extern "C" int32_t LIB$FIT_NODENAME(
/* |VR| %C_S|          */ DXX& NODENAME ,
/* |VR| VL| %C_S|      */ DXX& OUTPUT_BUFFER ,
/* |VR|           */ uint16_t* OUTPUT_WIDTH _Imm(0),
/* |VR| VL|           */ uint16_t* RESULTANT_LENGTH _Imm(0) )
;

extern "C" int32_t LIB$FIXUP_FLT(
/* |VR| US| %REF|      */ void* SIGNAL_ARGUMENTS ,
/* |VR| US| %REF|      */ void* MECHANISM_ARGUMENTS ,
/* |VR|           */ float* NEW_OPERAND _Imm(0) );

extern "C" int32_t LIB$FLT_UNDER(
/* |VR|           */ uint32_t* NEW_SETTING );

extern "C" int32_t LIB$FORMAT_DATE_TIME(
/* |VR| VL| %C_S|      */ DXX& DATE_STRING ,
/* |VR|           */ void* DATE _Imm(0),
/* |VR| VL|           */ uint32_t* USER_CONTEXT _Imm(0),
/* |VR| VL|           */ int32_t* DATE_LENGTH _Imm(0),
/* |VR|           */ uint32_t* FLAGS _Imm(0) );

```

```

%REF directory_id : ARRAY [$15..$u5:INTEGER] OF $UWORD :=
%IMMED 0;
VAR acp_status : [VOLATILE] UNSIGNED := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$file_scan (
%REF fab : [UNSAFE] ARRAY [$11..$u1:INTEGER] OF $UBYTE;
%IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_success_procedure;
%IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_error_procedure;
VAR context : [VOLATILE] UNSIGNED := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$file_scan_end (
%REF fab : [VOLATILE,UNSAFE] ARRAY [$11..$u1:INTEGER] OF
$UBYTE := %IMMED 0;
VAR context : [VOLATILE] UNSIGNED := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$find_file (
filespec : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
CHAR;
VAR resultant_filespec : [CLASS_S,VOLATILE] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR;
VAR context : [VOLATILE] UNSIGNED;
default_filespec : [CLASS_S] PACKED ARRAY
[$14..$u4:INTEGER] OF CHAR := %IMMED 0;
related_filespec : [CLASS_S] PACKED ARRAY
[$15..$u5:INTEGER] OF CHAR := %IMMED 0;
VAR status_value : [VOLATILE] UNSIGNED := %IMMED 0;
flags : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$find_file_end (
context : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$find_image_symbol (
filename : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
CHAR;
symbol : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER] OF
CHAR;
VAR symbol_value : [VOLATILE] INTEGER;
image_name : [CLASS_S] PACKED ARRAY [$14..$u4:INTEGER] OF
CHAR := %IMMED 0;
%IMMED flags : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

```

```

extern "C" int32_t LIB$FORMAT_SOGW_PROT(
/* |VR| */ uint16_t* PROTECTION_MASK ,
/* |VR| %REF| */ void* ACCESS_NAMES _Imm(0),
/* |VR| %REF| */ void* OWNERSHIP_NAMES _Imm(0),
/* |VR| %C_S| */ DXX& OWNERSHIP_SEPARATOR
_RefZero(),
/* |VR| %C_S| */ DXX& LIST_SEPARATOR _RefZero(),
/* |VR| VL| %C_S| */ DXX& PROTECTION_STRING _RefZero(),
/* |VR| VL| */ int16_t* PROTECTION_LENGTH _Imm(0))
;

extern "C" int32_t LIB$FREE_DATE_TIME_CONTEXT(
/* |VR| VL| */ uint32_t* USER_CONTEXT _Imm(0)) ;

extern "C" int32_t LIB$FREE_EF(
/* |VR| */ uint32_t* EVENT_FLAG_NUMBER ) ;

extern "C" int32_t LIB$FREE_LUN(
/* |VR| */ int32_t* LOGICAL_UNIT_NUMBER ) ;

extern "C" int32_t LIB$FREE_TIMER(
/* |VR| VL| */ $DEFTYP* HANDLE_ADDRESS ) ;

extern "C" int32_t LIB$FREE_VM(
/* |VR| */ int32_t* NUMBER_OF_BYTES ,
/* |VR| */ $DEFTYP* BASE_ADDRESS ,
/* |VR| */ uint32_t* ZONE_ID _Imm(0)) ;

extern "C" int32_t LIB$FREE_VM_64(
/* |VR| */ void* NUMBER_OF_BYTES ,
/* |VR| */ void* BASE_ADDRESS ,
/* |VR| */ void* ZONE_ID _Imm(0)) ;

extern "C" int32_t LIB$FREE_VM_PAGE(
/* |VR| */ int32_t* NUMBER_OF_PAGES ,
/* |VR| */ $DEFTYP* BASE_ADDRESS ) ;

extern "C" int32_t LIB$FREE_VM_PAGE_64(
/* |VR| */ void* NUMBER_OF_PAGES ,
/* |VR| */ void* BASE_ADDRESS ) ;

```

```

[ASYNCHRONOUS] FUNCTION lib$find_vm_zone (
    VAR context : [VOLATILE] UNSIGNED;
    VAR zone_id : [VOLATILE] UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$find_vm_zone_64 (
    VAR context : [VOLATILE] $UQUAD;
    VAR zone_id : [VOLATILE] $UQUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$fit_nodename (
    nodename : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
CHAR;
    VAR output_buffer : [CLASS_S,VOLATILE] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR;
    output_width : $UWORD := %IMMED 0;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$fixup_flt (
    %REF signal_arguments : [UNSAFE] ARRAY [$11..$u1:INTEGER]
OF $SUBYTE;
    %REF mechanism_arguments : [UNSAFE] ARRAY
[$12..$u2:INTEGER] OF $SUBYTE;
    new_operand : SINGLE := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$flt_under (
    new_setting : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$format_date_time (
    VAR date_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
    date : $UQUAD := %IMMED 0;
    VAR user_context : [VOLATILE] UNSIGNED := %IMMED 0;
    VAR date_length : [VOLATILE] INTEGER := %IMMED 0;
    flags : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$format_sogw_prot (
    protection_mask : $UWORD;
    %REF access_names : ARRAY [$12..$u2:INTEGER] OF $UQUAD :=
%IMMED 0;
    %REF ownership_names : ARRAY [$13..$u3:INTEGER] OF $UQUAD
:= %IMMED 0;
    ownership_separator : [CLASS_S] PACKED ARRAY
[$14..$u4:INTEGER] OF CHAR := %IMMED 0;
    list_separator : [CLASS_S] PACKED ARRAY
[$15..$u5:INTEGER] OF CHAR := %IMMED 0;

```

```

extern "C" int32_t LIB$GETDVI(
/* |VR| */ int32_t* ITEM_CODE ,
/* |VR| */ uint16_t* CHANNEL _Imm(0),
/* |VR| %C_S| DXX& DEVICE_NAME _RefZero(),
/* |VR| VL| int32_t* LONGWORD_INTEGER_VALUE
_Imm(0),
/* |VR| VL| %C_S| DXX& RESULTANT_STRING _RefZero(),
/* |VR| VL| uint16_t* RESULTANT_LENGTH _Imm(0))
;

extern "C" int32_t LIB$GETJPI(
/* |VR| */ int32_t* ITEM_CODE ,
/* |VR| VL| uint32_t* PROCESS_ID _Imm(0),
/* |VR| %C_S| DXX& PROCESS_NAME _RefZero(),
/* |VR| US| VL| %REF| void* RESULTANT_VALUE _Imm(0),
/* |VR| VL| %C_S| DXX& RESULTANT_STRING _RefZero(),
/* |VR| VL| uint16_t* RESULTANT_LENGTH _Imm(0))
;

extern "C" int32_t LIB$GETQUI(
/* |VR| */ int32_t* FUNCTION_CODE ,
/* |VR| */ int32_t* ITEM_CODE _Imm(0),
/* |VR| */ int32_t* SEARCH_NUMBER _Imm(0),
/* |VR| %C_S| DXX& SEARCH_NAME _RefZero(),
/* |VR| */ uint32_t* SEARCH_FLAGS _Imm(0),
/* |VR| US| VL| %REF| void* RESULTANT_VALUE _Imm(0),
/* |VR| VL| %C_S| DXX& RESULTANT_STRING _RefZero(),
/* |VR| VL| uint16_t* RESULTANT_LENGTH _Imm(0))
;

extern "C" int32_t LIB$GETSYI(
/* |VR| */ int32_t* ITEM_CODE ,
/* |VR| US| VL| %REF| void* RESULTANT_VALUE _Imm(0),
/* |VR| VL| %C_S| DXX& RESULTANT_STRING _RefZero(),
/* |VR| VL| uint16_t* RESULTANT_LENGTH _Imm(0),
/* |VR| VL| uint32_t* CLUSTER_SYSTEM_ID
_Imm(0),
/* |VR| %C_S| DXX& NODE_NAME _RefZero());

extern "C" int32_t LIB$GET_ACCNAM(
/* |VR| %C_S| DXX& CLSNAM _RefZero(),
/* |VR| %C_S| DXX& OBJNAM _RefZero(),
/* |VR| VL| uint32_t* ACCNAM _Imm(0));

extern "C" int32_t LIB$GET_ACCNAM_BY_CONTEXT(
/* |VR| */ uint32_t* CONTEXT ,
/* |VR| VL| uint32_t* ACCNAM );

```

```

VAR protection_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$16..$u6:INTEGER] OF CHAR;
VAR protection_length : [VOLATILE] $WORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$free_date_time_context (
    VAR user_context : [VOLATILE] UNSIGNED := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$free_ef (
    event_flag_number : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$free_lun (
    logical_unit_number : INTEGER) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$free_timer (
    VAR handle_address : [VOLATILE] $DEFPTR) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$free_vm (
    number_of_bytes : INTEGER;
    base_address : $DEFPTR;
    zone_id : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$free_vm_64 (
    number_of_bytes : $QUAD;
    base_address : lib$routines$$$typ2;
    zone_id : $UQUAD := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$free_vm_page (
    number_of_pages : INTEGER;
    base_address : $DEFPTR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$free_vm_page_64 (
    number_of_pages : $QUAD;
    base_address : lib$routines$$$typ3) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$getdvi (
    item_code : INTEGER;
    channel : $UWORD := %IMMED 0;
    device_name : [CLASS_S] PACKED ARRAY [$13..$u3:INTEGER]
OF CHAR := %IMMED 0;

```

```

extern "C" int32_t LIB$GET_COMMAND(
/* |VR|VL|%C_S|          */ DXX& RESULTANT_STRING ,
/* |VR|%C_S|            */ DXX& PROMPT_STRING_RefZero(),
/* |VR|VL|              */ uint16_t* RESULTANT_LENGTH _Imm(0))
;

extern "C" int32_t LIB$GET_COMMON(
/* |VR|VL|%C_S|          */ DXX& RESULTANT_STRING ,
/* |VR|VL|              */ uint16_t* RESULTANT_LENGTH _Imm(0))
;

extern "C" void LIB$GET_CURR_INVO_CONTEXT(
/* |VR|VL|              */ $DEFTYP* INVO_CONTEXT ) ;

extern "C" int32_t LIB$GET_CURR_INVO_HANDLE(void) ;

extern "C" int32_t LIB$GET_DATE_FORMAT(
/* |VR|VL|%C_S|          */ DXX& FORMAT_STRING ,
/* |VR|VL|              */ uint32_t* USER_CONTEXT _Imm(0)) ;

extern "C" int32_t LIB$GET_EF(
/* |VR|VL|              */ uint32_t* EVENT_FLAG_NUMBER ) ;

extern "C" int32_t LIB$GET_FOREIGN(
/* |VR|VL|%C_S|          */ DXX& RESULTANT_STRING ,
/* |VR|%C_S|            */ DXX& PROMPT_STRING_RefZero(),
/* |VR|VL|              */ uint16_t* RESULTANT_LENGTH _Imm(0),
/* |VR|VL|              */ uint32_t* FLAGS _Imm(0)) ;

extern "C" int32_t LIB$GET_FULLNAME_OFFSET(
/* |VR|%C_S|            */ DXX& FULLNAME ,
/* |VR|VL|              */ uint16_t* OFFSET ) ;

extern "C" int32_t LIB$GET_HOSTNAME(
/* |VR|VL|%C_S|          */ DXX& HOSTNAME ,
/* |VR|VL|              */ uint16_t* RESULTANT_LENGTH _Imm(0),
/* |VR|              */ uint32_t* FLAGS _Imm(0)) ;

extern "C" int32_t LIB$GET_INPUT(
/* |VR|VL|%C_S|          */ DXX& RESULTANT_STRING ,
/* |VR|%C_S|            */ DXX& PROMPT_STRING_RefZero(),

```



```

VAR longword_integer_value : [VOLATILE] INTEGER := %IMMED
0;
VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$15..$u5:INTEGER] OF CHAR := %IMMED 0;
VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$getjpi (
item_code : INTEGER;
VAR process_id : [VOLATILE] UNSIGNED := %IMMED 0;
process_name : [CLASS_S] PACKED ARRAY [$13..$u3:INTEGER]
OF CHAR := %IMMED 0;
%REF resultant_value : [VOLATILE,UNSAFE] ARRAY
[$14..$u4:INTEGER] OF $UBYTE := %IMMED 0;
VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$15..$u5:INTEGER] OF CHAR := %IMMED 0;
VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$getqui (
function_code : INTEGER;
item_code : INTEGER := %IMMED 0;
search_number : INTEGER := %IMMED 0;
search_name : [CLASS_S] PACKED ARRAY [$14..$u4:INTEGER]
OF CHAR := %IMMED 0;
search_flags : UNSIGNED := %IMMED 0;
%REF resultant_value : [VOLATILE,UNSAFE] ARRAY
[$16..$u6:INTEGER] OF $UBYTE := %IMMED 0;
VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$17..$u7:INTEGER] OF CHAR := %IMMED 0;
VAR resultant_length : [VOLATILE] $WORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$getsysi (
item_code : INTEGER;
%REF resultant_value : [VOLATILE,UNSAFE] ARRAY
[$12..$u2:INTEGER] OF $UBYTE := %IMMED 0;
VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$13..$u3:INTEGER] OF CHAR := %IMMED 0;
VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0;
VAR cluster_system_id : [VOLATILE] UNSIGNED := %IMMED 0;
node_name : [CLASS_S] PACKED ARRAY [$16..$u6:INTEGER] OF
CHAR := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_accnam (
clsnam : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
CHAR := %IMMED 0;
/* |VR|VL| */ uint16_t* RESULTANT_LENGTH _Imm(0)
;

extern "C" int32_t LIB$GET_INVO_CONTEXT(
/* |PR|%IMM| */ uint32_t INVO_HANDLE ,
/* |VR|VL| */ $DEFTYP* INVO_CONTEXT ) ;

extern "C" int32_t LIB$GET_INVO_HANDLE(
/* |VR|VL| */ $DEFTYP* INVO_CONTEXT ) ;

extern "C" int32_t LIB$GET_LOGICAL(
/* |VR| %C_S| */ DXX& LOGICAL_NAME ,
/* |VR|VL| %C_S| */ DXX& RESULTANT_STRING_RefZero(),
/* |VR|VL| */ uint16_t* RESULTANT_LENGTH _Imm(0),
/* |VR| %C_S| */ DXX& TABLE_NAME_RefZero(),
/* |VR|VL| */ int32_t* MAX_INDEX _Imm(0),
/* |VR| */ uint32_t* INDEX _Imm(0),
/* |VR| */ uchar_t* ACMODE _Imm(0),
/* |VR| */ uint32_t* FLAGS _Imm(0)) ;

extern "C" int32_t LIB$GET_LUN(
/* |VR|VL| */ int32_t* LOGICAL_UNIT_NUMBER ) ;

extern "C" int32_t LIB$GET_MAXIMUM_DATE_LENGTH(
/* |VR|VL| */ int32_t* DATE_LENGTH ,
/* |VR|VL| */ uint32_t* USER_CONTEXT _Imm(0),
/* |VR| */ uint32_t* FLAGS _Imm(0)) ;

extern "C" int32_t LIB$GET_PREV_INVO_CONTEXT(
/* |VR|VL| */ $DEFTYP* INVO_CONTEXT ) ;

extern "C" int32_t LIB$GET_PREV_INVO_HANDLE(
/* |PR|%IMM| */ uint32_t INVO_HANDLE ) ;

extern "C" int32_t LIB$GET_SYMBOL(
/* |VR| %C_S| */ DXX& SYMBOL ,
/* |VR|VL| %C_S| */ DXX& RESULTANT_STRING ,
/* |VR|VL| */ uint16_t* RESULTANT_LENGTH _Imm(0),
/* |VR|VL| */ int32_t* TABLE_TYPE_INDICATOR
_Imm(0)) ;

extern "C" int32_t LIB$GET_USERS_LANGUAGE(
/* |VR|VL| %C_S| */ DXX& LANGUAGE ) ;

```



```

VAR flags : [VOLATILE] UNSIGNED := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_fullname_offset (
    fullname : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
    CHAR;
    VAR offset : [VOLATILE] $UWORD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_hostname (
    VAR hostname : [CLASS_S,VOLATILE] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0;
    flags : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_input (
    VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
    prompt_string : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER]
    OF CHAR := %IMMED 0;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
    INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_invo_context (
    %IMMED invo_handle : UNSIGNED;
    VAR invo_context : [VOLATILE] $DEFPTR) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_invo_handle (
    VAR invo_context : [VOLATILE] $DEFPTR) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_logical (
    logical_name : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
    OF CHAR;
    VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR := %IMMED 0;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0;
    table_name : [CLASS_S] PACKED ARRAY [$14..$u4:INTEGER] OF
    CHAR := %IMMED 0;
    VAR max_index : [VOLATILE] INTEGER := %IMMED 0;
    index : UNSIGNED := %IMMED 0;
    acmode : $BYTE := %IMMED 0;
    flags : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_lun (

```

```

extern "C" int32_t LIB$INSERT_TREE_64(
/* |VR|VL| */ void* TREEHEAD ,
/* |VR|US|REF| */ void* SYMBOL ,
/* |VR| */ uint32_t* FLAGS ,
/* |PR|PROC|BO|AS|IMM| */ uint32_t USER_COMPARE_ROUTINE ,
/* |PR|PROC|BO|AS|IMM| */ uint32_t USER_ALLOCATION_PROCEDURE
,
/* |VR|VL| */ void* NEW_NODE ,
/* |PR|IMM| */ int32_t USER_DATA = 0) ;

extern "C" int32_t LIB$INSQHI(
/* |VR|US|VL|REF| */ void* ENTRY ,
/* |VR|VL| */ void* HEADER ,
/* |VR| */ uint32_t* RETRY_COUNT _Imm(0) ) ;

extern "C" int32_t LIB$INSQHIQ(
/* |VR|US|VL|REF| */ void* ENTRY ,
/* |VR|VL| */ void* HEADER ,
/* |VR| */ uint32_t* RETRY_COUNT _Imm(0) ) ;

extern "C" int32_t LIB$INSQTI(
/* |VR|US|VL|REF| */ void* ENTRY ,
/* |VR|VL| */ void* HEADER ,
/* |VR| */ uint32_t* RETRY_COUNT _Imm(0) ) ;

extern "C" int32_t LIB$INSQTIQ(
/* |VR|US|VL|REF| */ void* ENTRY ,
/* |VR|VL| */ void* HEADER ,
/* |VR| */ uint32_t* RETRY_COUNT _Imm(0) ) ;

extern "C" void LIB$INSV(
/* |VR| */ int32_t* LONGWORD_INTEGER_SOURCE ,
/* |VR| */ int32_t* POSITION ,
/* |VR| */ uchar_t* SIZE ,
/* |PR|IMM| */ $DEFTYP BASE_ADDRESS ) ;

extern "C" int32_t LIB$INT_OVER(
/* |VR| */ uint32_t* NEW_SETTING ) ;

extern "C" uint16_t LIB$LEN(
/* |VR|C_S| */ DXX& SOURCE_STRING ) ;

```

```

VAR logical_unit_number : [VOLATILE] INTEGER) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_maximum_date_length (
    VAR date_length : [VOLATILE] INTEGER;
    VAR user_context : [VOLATILE] UNSIGNED := %IMMED 0;
    flags : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_prev_invo_context (
    VAR invo_context : [VOLATILE] $DEFPTR) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_prev_invo_handle (
    %IMMED invo_handle : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_symbol (
    symbol : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER] OF
    CHAR;
    VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
    [$l2..$u2:INTEGER] OF CHAR;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0;
    VAR table_type_indicator : [VOLATILE] INTEGER := %IMMED
    0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_users_language (
    VAR language : [CLASS_S,VOLATILE] PACKED ARRAY
    [$l1..$u1:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_vm (
    number_of_bytes : INTEGER;
    VAR base_address : [VOLATILE] $DEFPTR;
    zone_id : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_vm_64 (
    number_of_bytes : $QUAD;
    VAR base_address : [VOLATILE] lib$routines$$typ4;
    zone_id : $UQUAD := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$get_vm_page (
    number_of_pages : INTEGER;
    VAR base_address : [VOLATILE] $DEFPTR) : INTEGER;
EXTERNAL;

```

```

extern "C" int32_t LIB$LOCC(
/* |VR| %C_S| */ DXX& CHARACTER_STRING ,
/* |VR| %C_S| */ DXX& SOURCE_STRING ) ;

extern "C" int32_t LIB$LOOKUP_KEY(
/* |VR| %C_S| */ DXX& SEARCH_STRING ,
/* |VR| US| %REF| */ void* KEY_TABLE_ARRAY ,
/* |VR| VL| */ uint32_t* KEY_VALUE _Imm(0),
/* |VR| VL| %C_S| */ DXX& KEYWORD_STRING _RefZero(),
/* |VR| VL| */ uint16_t* RESULTANT_LENGTH _Imm(0))
;

extern "C" int32_t LIB$LOOKUP_TREE(
/* |VR| */ $DEFTYP* TREEHEAD ,
/* |VR| US| %REF| */ void* SYMBOL ,
/* |PR| PROC|BO|AS| %IMM| */ uint32_t USER_COMPARE_ROUTINE ,
/* |VR| VL| */ $DEFTYP* NEW_NODE ) ;

extern "C" int32_t LIB$LOOKUP_TREE_64(
/* |VR| */ void* TREEHEAD ,
/* |VR| US| %REF| */ void* SYMBOL ,
/* |PR| PROC|BO|AS| %IMM| */ uint32_t USER_COMPARE_ROUTINE ,
/* |VR| VL| */ void* NEW_NODE ) ;

extern "C" int32_t LIB$LP_LINES(void) ;

extern "C" int32_t LIB$MATCHC(
/* |VR| %C_S| */ DXX& SUB_STRING ,
/* |VR| %C_S| */ DXX& SOURCE_STRING ) ;

extern "C" int32_t LIB$MATCH_COND(
/* |VR| */ uint32_t* MATCH_CONDITION_VALUE ,
/* |VR| */ uint32_t* COMPARE_CONDITION_VALUE ,
/* |LS| */ int32_t _ListArgs = 0, ...)
;

extern "C" void LIB$MOVC3(
/* |VR| */ uint16_t* WORD_INTEGER_LENGTH ,
/* |VR| US| %REF| */ void* SOURCE ,
/* |VR| US| VL| %REF| */ void* DESTINATION ) ;

extern "C" void LIB$MOVC5(
/* |VR| */ uint16_t*
WORD_INTEGER_SOURCE_LENGTH ,

```

```

[ASYNCHRONOUS] FUNCTION lib$get_vm_page_64 (
    number_of_pages : $QUAD;
    VAR base_address : [VOLATILE] lib$routines$$$typ5) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$ichar (
    source_string : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER]
OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$index (
    source_string : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER]
OF CHAR;
    sub_string : [CLASS_S] PACKED ARRAY [$l2..$u2:INTEGER] OF
CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$init_date_time_context (
    VAR user_context : [VOLATILE] UNSIGNED;
    component : INTEGER;
    init_string : [CLASS_S] PACKED ARRAY [$l3..$u3:INTEGER]
OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$init_timer (
    VAR context : [VOLATILE] UNSIGNED := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$insert_tree (
    VAR treehead : [VOLATILE] $DEFPTR;
    %REF symbol : [UNSAFE] ARRAY [$l2..$u2:INTEGER] OF
$UBYTE;
    flags : UNSIGNED;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_compare_routine;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_allocation_procedure;
    VAR new_node : [VOLATILE] $DEFPTR;
    %IMMED user_data : UNSIGNED := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$insert_tree_64 (
    VAR treehead : [VOLATILE] lib$routines$$$typ6;
    %REF symbol : [UNSAFE] ARRAY [$l2..$u2:INTEGER] OF
$UBYTE;
    flags : UNSIGNED;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_compare_routine;

```

```

/* |VR|US|%REF|          */ void* SOURCE ,
/* |VR|                  */ char* FILL ,
/* |VR|                  */ uint16_t*
WORD_INTEGER_DESTINATION_LENGTH ,
/* |VR|US|VL|%REF|      */ void* DESTINATION ) ;

extern "C" int32_t LIB$MOVTC(
/* |VR| %C_S|            */ DXX& SOURCE_STRING ,
/* |VR| %C_S|            */ DXX& FILL_CHARACTER ,
/* |VR| %C_S|            */ DXX& TRANSLATION_TABLE ,
/* |VR| VL| %C_S|        */ DXX& DESTINATION_STRING ) ;

extern "C" int32_t LIB$MOVTC(
/* |VR| %C_S|            */ DXX& SOURCE_STRING ,
/* |VR| %C_S|            */ DXX& STOP_CHARACTER ,
/* |VR| %C_S|            */ DXX& TRANSLATION_TABLE ,
/* |VR| VL| %C_S|        */ DXX& DESTINATION_STRING ,
/* |VR| %C_S|            */ DXX& FILL_CHARACTER_RefZero() ) ;

extern "C" int32_t LIB$MT_DENSITY_CONVERSION(
/* |PR| %IMM|            */ uint32_t DEVICE_CHANNEL ,
/* |VR| VL|              */ uint32_t* MT_DENSITY_SYMBOL ,
/* |VR| VL|              */ uint32_t* MT3_ENHANCED_DRIVER ) ;

extern "C" int32_t LIB$MT_DENSITY_VERIFICATION(
/* |PR| %IMM|            */ uint32_t DEVICE_CHANNEL ,
/* |PR| %IMM|            */ uint32_t MT_DENSITY_SYMBOL ,
/* |VR| VL|              */ uint32_t* MT3_ENHANCED_DRIVER ) ;

extern "C" int32_t LIB$MT_DENSITY_STRCVT(
/* |PR| %IMM|            */ uint32_t MT_DENSITY_SYMBOL ,
/* |VR| VL| %REF|        */ DXX& OUT_STRING ) ;

extern "C" int32_t LIB$MULTF_DELTA_TIME(
/* |VR|                  */ float* MULTIPLIER ,
/* |VR| VL|              */ void* DELTA_TIME ) ;

extern "C" int32_t LIB$MULT_DELTA_TIME(
/* |VR|                  */ int32_t* MULTIPLIER ,
/* |VR| VL|              */ void* DELTA_TIME ) ;

extern "C" int32_t LIB$PARSE_ACCESS_CODE(
/* |VR| %C_S|            */ DXX& ACCESS_STRING ,
/* |VR| %REF|            */ void* ACCESS_NAMES_Imm(0),

```

```

    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_allocation_procedure;
    VAR new_node : [VOLATILE] lib$routines$$styp7;
    %IMMED user_data : INTEGER := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$insqhi (
    %REF ENTRY : [VOLATILE,UNSAFE] ARRAY [$l1..$u1:INTEGER]
OF $SUBYTE;
    VAR header : [VOLATILE] $QUAD;
    retry_count : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$insqhiq (
    %REF ENTRY : [VOLATILE,UNSAFE] ARRAY [$l1..$u1:INTEGER]
OF $SUBYTE;
    VAR header : [VOLATILE] $OCTA;
    retry_count : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$insqti (
    %REF ENTRY : [VOLATILE,UNSAFE] ARRAY [$l1..$u1:INTEGER]
OF $SUBYTE;
    VAR header : [VOLATILE] $QUAD;

    retry_count : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$insqtiq (
    %REF ENTRY : [VOLATILE,UNSAFE] ARRAY [$l1..$u1:INTEGER]
OF $SUBYTE;
    VAR header : [VOLATILE] $OCTA;
    retry_count : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] PROCEDURE lib$insv (
    longword_integer_source : INTEGER;
    position : INTEGER;
    size : $SUBYTE;
    %IMMED base_address : $DEFPTR); EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$int_over (
    new_setting : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$len (
    source_string : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER]
OF CHAR) : $UWORD; EXTERNAL;

```

```

/* |VR|                               */ uint16_t* OWNERSHIP_CATEGORY
   _Imm(0),
/* |VR|VL|                             */ uint16_t* ACCESS_MASK _Imm(0),
/* |VR|VL|                             */ int16_t* END_POSITION _Imm(0)) ;

extern "C" int32_t LIB$PARSE_SOGW_PROT(
/* |VR| %C_S|                           */ DXX& PROTECTION_STRING ,
/* |VR| %REF|                             */ void* ACCESS_NAMES _Imm(0),
/* |VR| VL|                               */ uint16_t* PROTECTION_MASK _Imm(0),
/* |VR| VL|                               */ uint16_t* OWNERSHIP_MASK _Imm(0),
/* |VR| VL|                               */ int16_t* END_POSITION _Imm(0)) ;

extern "C" int32_t LIB$PAUSE(void) ;

extern "C" int32_t LIB$POLYD(
/* |VR|                               */ double* POLYNOMIAL_ARGUMENT ,
/* |VR|                               */ int16_t* DEGREE ,
/* |VR| %REF|                             */ void* COEFFICIENT ,
/* |VR| VL|                               */ double* FLOATING_POINT_RESULT ) ;

extern "C" int32_t LIB$POLYF(
/* |VR|                               */ float* POLYNOMIAL_ARGUMENT ,
/* |VR|                               */ int16_t* DEGREE ,
/* |VR| %REF|                             */ void* COEFFICIENT ,
/* |VR| VL|                               */ float* FLOATING_POINT_RESULT ) ;

extern "C" int32_t LIB$POLYG(
/* |VR|                               */ double* POLYNOMIAL_ARGUMENT ,
/* |VR|                               */ int16_t* DEGREE ,
/* |VR| %REF|                             */ void* COEFFICIENT ,
/* |VR| VL|                               */ double* FLOATING_POINT_RESULT ) ;

extern "C" int32_t LIB$POLYH(
/* |VR|                               */ int64_t* POLYNOMIAL_ARGUMENT ,
/* |VR|                               */ int16_t* DEGREE ,
/* |VR| %REF|                             */ void* COEFFICIENT ,
/* |VR| VL|                               */ int64_t* FLOATING_POINT_RESULT ) ;

extern "C" int32_t LIB$PUT_COMMON(
/* |VR| %C_S|                             */ DXX& SOURCE_STRING ,
/* |VR| VL|                               */ uint16_t* RESULTANT_LENGTH _Imm(0))
;

extern "C" int32_t LIB$PUT_INVO_REGISTERS(

```

```

[ASYNCHRONOUS] FUNCTION lib$locc (
    character_string : [CLASS_S] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
    source_string : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER]
OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$lookup_key (
    search_string : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
OF CHAR;
    %REF key_table_array : [UNSAFE] ARRAY [$12..$u2:INTEGER]
OF $SUBYTE;
    VAR key_value : [VOLATILE] UNSIGNED := %IMMED 0;
    VAR keyword_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$14..$u4:INTEGER] OF CHAR := %IMMED 0;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$lookup_tree (
    treehead : $DEFPTR;
    %REF symbol : [UNSAFE] ARRAY [$12..$u2:INTEGER] OF
$SUBYTE;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_compare_routine;
    VAR new_node : [VOLATILE] $DEFPTR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$lookup_tree_64 (
    treehead : lib$routines$$typ8;
    %REF symbol : [UNSAFE] ARRAY [$12..$u2:INTEGER] OF
$SUBYTE;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_compare_routine;
    VAR new_node : [VOLATILE] lib$routines$$typ9) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$lp_lines : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$matchc (
    sub_string : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
CHAR;
    source_string : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER]
OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$match_cond (
    match_condition_value : UNSIGNED;
    compare_condition_value : UNSIGNED;
    $p3 : [LIST] UNSIGNED) : INTEGER; EXTERNAL;

```

```

/* |PR|%IMM|          */ uint32_t INVO_HANDLE ,
/* |VR|              */ $DEFTYP* INVO_CONTEXT ,
/* |VR|              */ void* INVO_MASK ) ;

extern "C" int32_t LIB$PUT_OUTPUT(
/* |VR| %C_S|        */ DXX& MESSAGE_STRING ) ;

extern "C" int32_t LIB$RADIX_POINT(
/* |VR| VL| %C_S|    */ DXX& RADIX_POINT_STRING ,
/* |VR| VL|          */ uint16_t* RESULTANT_LENGTH _Imm(0))
;

extern "C" int32_t LIB$REMQUI(
/* |VR| VL|         */ void* HEADER ,
/* |VR| VL|         */ $DEFTYP* REMQUE_ADDRESS ,
/* |VR|           */ uint32_t* RETRY_COUNT _Imm(0)) ;

extern "C" int32_t LIB$REMQUIQ(
/* |VR| VL|         */ void* HEADER ,
/* |VR| VL|         */ void* REMQUE_ADDRESS ,
/* |VR|           */ uint32_t* RETRY_COUNT _Imm(0)) ;

extern "C" int32_t LIB$REMQTI(
/* |VR| VL|         */ void* HEADER ,
/* |VR| VL|         */ $DEFTYP* REMQUE_ADDRESS ,
/* |VR|           */ uint32_t* RETRY_COUNT _Imm(0)) ;

extern "C" int32_t LIB$REMQTIQ(
/* |VR| VL|         */ void* HEADER ,
/* |VR| VL|         */ void* REMQUE_ADDRESS ,
/* |VR|           */ uint32_t* RETRY_COUNT _Imm(0)) ;

extern "C" int32_t LIB$RENAME_FILE(
/* |VR| %C_S|        */ DXX& OLD_FILESPEC ,
/* |VR| %C_S|        */ DXX& NEW_FILESPEC ,
/* |VR| %C_S|        */ DXX& DEFAULT_FILESPEC_RefZero(),
/* |VR| %C_S|        */ DXX& RELATED_FILESPEC_RefZero(),
/* |VR|             */ uint32_t* FLAGS _Imm(0),
/* |PR| PROC|BO|AS| %IMM| */ uint32_t USER_SUCCESS_PROCEDURE =
0,
/* |PR| PROC|BO|AS| %IMM| */ uint32_t USER_ERROR_PROCEDURE = 0,
/* |PR| PROC|BO|AS| %IMM| */ uint32_t USER_CONFIRM_PROCEDURE =
0,
/* |PR| %IMM|         */ uint32_t USER_SPECIFIED_ARGUMENT =
0,

```

```

[ASYNCHRONOUS] PROCEDURE lib$movc3 (
    word_integer_length : $UWORD;
    %REF source : [UNSAFE] ARRAY [$12..$u2:INTEGER] OF
$SUBYTE;
    %REF destination : [VOLATILE,UNSAFE] ARRAY
[$13..$u3:INTEGER] OF $SUBYTE); EXTERNAL;

[ASYNCHRONOUS] PROCEDURE lib$movc5 (
    word_integer_source_length : $UWORD;
    %REF source : [UNSAFE] ARRAY [$12..$u2:INTEGER] OF
$SUBYTE;
    FILL : $BYTE;
    word_integer_destination_length : $UWORD;
    %REF destination : [VOLATILE,UNSAFE] ARRAY
[$15..$u5:INTEGER] OF $SUBYTE); EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$movtc (
    source_string : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
OF CHAR;
    fill_character : [CLASS_S] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR;
    translation_table : [CLASS_S] PACKED ARRAY
[$13..$u3:INTEGER] OF CHAR;
    VAR destination_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$14..$u4:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$movtuc (
    source_string : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
OF CHAR;
    stop_character : [CLASS_S] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR;
    translation_table : [CLASS_S] PACKED ARRAY
[$13..$u3:INTEGER] OF CHAR;
    VAR destination_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$14..$u4:INTEGER] OF CHAR;
    fill_character : [CLASS_S] PACKED ARRAY
[$15..$u5:INTEGER] OF CHAR := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$mt_density_conversion (
    %IMMED device_channel : UNSIGNED;
    VAR mt_density_symbol : [VOLATILE] UNSIGNED;
    VAR mt3_enhanced_driver : [VOLATILE] UNSIGNED) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$mt_density_verification (

```

```

/* |VR|VL|%C_S|          */ DXX& OLD_RESULTANT_NAME _RefZero(),
/* |VR|VL|%C_S|          */ DXX& NEW_RESULTANT_NAME _RefZero(),
/* |VR|VL|              */ uint32_t* FILE_SCAN_CONTEXT
_Imm(0) );

extern "C" int32_t LIB$RESERVE_EF(
/* |VR|                  */ uint32_t* EVENT_FLAG_NUMBER );

extern "C" int32_t LIB$RESET_VM_ZONE(
/* |VR|                  */ uint32_t* ZONE_ID );

extern "C" int32_t LIB$RESET_VM_ZONE_64(
/* |VR|                  */ void* ZONE_ID );

extern "C" int32_t LIB$RUN_PROGRAM(
/* |VR|%C_S|             */ DXX& PROGRAM_NAME );

extern "C" int32_t LIB$SCANC(
/* |VR|%C_S|             */ DXX& SOURCE_STRING ,
/* |VR|%REF|             */ void* TABLE_ARRAY ,
/* |VR|                  */ uchar_t* BYTE_INTEGER_MASK );

extern "C" int32_t LIB$SCOPY_DXD(
/* |VR|%C_S|             */ DXX& SOURCE_STRING ,
/* |VR|VL|%C_S|         */ DXX& DESTINATION_STRING );

extern "C" int32_t LIB$SCOPY_R_DX(
/* |VR|                  */ uint16_t*
WORD_INTEGER_SOURCE_LENGTH ,
/* |VR|%REF|             */ DXX& SOURCE_STRING ,
/* |VR|VL|%C_S|         */ DXX& DESTINATION_STRING );

extern "C" int32_t LIB$SCOPY_R_DX_64(
/* |VR|                  */ void* QUAD_INTEGER_SOURCE_LENGTH ,
/* |VR|%REF|             */ DXX& SOURCE_STRING ,
/* |VR|VL|%C_S|         */ DXX& DESTINATION_STRING );

extern "C" int32_t LIB$SET_LOGICAL(
/* |VR|%C_S|             */ DXX& LOGICAL_NAME ,
/* |VR|%C_S|             */ DXX& VALUE_STRING _RefZero(),
/* |VR|%C_S|             */ DXX& TABLE _RefZero(),
/* |VR|                  */ uint32_t* ATTRIBUTES _Imm(0),
/* |VR|US|%REF|          */ void* ITEM_LIST _Imm(0) );

```



```

    %IMMED device_channel : UNSIGNED;
    %IMMED mt_density_symbol : UNSIGNED;
    VAR mt3_enhanced_driver : [VOLATILE] UNSIGNED) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$mt_density_strcvrt (
    %IMMED mt_density_symbol : UNSIGNED;
    %REF out_string : [VOLATILE] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$multf_delta_time (
    multiplier : SINGLE;
    VAR delta_time : [VOLATILE] $UQUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$mult_delta_time (
    multiplier : INTEGER;
    VAR delta_time : [VOLATILE] $UQUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$parse_access_code (
    access_string : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
OF CHAR;
    %REF access_names : ARRAY [$12..$u2:INTEGER] OF $UQUAD :=
%IMMED 0;
    ownership_category : $UWORD;
    VAR access_mask : [VOLATILE] $UWORD;
    VAR end_position : [VOLATILE] $WORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$parse_sogw_prot (
    protection_string : [CLASS_S] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
    %REF access_names : ARRAY [$12..$u2:INTEGER] OF $UQUAD :=
%IMMED 0;
    VAR protection_mask : [VOLATILE] $UWORD;
    VAR ownership_mask : [VOLATILE] $UWORD;
    VAR end_position : [VOLATILE] $WORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$pause : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$polyd (
    polynomial_argument : D_FLOAT$$TYPE;
    degree : $WORD;
    %REF coefficient : ARRAY [$13..$u3:INTEGER] OF
D_FLOAT$$TYPE;

```

```

extern "C" int32_t LIB$SET_SYMBOL(
/* |VR| %C_S| */ DXX& SYMBOL ,
/* |VR| %C_S| */ DXX& VALUE_STRING ,
/* |VR| */ int32_t* TABLE_TYPE_INDICATOR
_Imm(0) ) ;

extern "C" int32_t LIB$SFREE1_DD(
/* |VR|VL| */ void* DESCRIPTOR_ADDRESS ) ;

extern "C" int32_t LIB$SFREEN_DD(
/* |VR| */ uint32_t* NUMBER_OF_DESCRIPTOR ,
/* |VR|VL| %REF| */ void* FIRST_DESCRIPTOR_ARRAY ) ;

extern "C" int32_t LIB$SGET1_DD(
/* |VR| */ uint16_t* WORD_INTEGER_LENGTH ,
/* |VR|VL| */ void* DESCRIPTOR_PART ) ;

extern "C" int32_t LIB$SGET1_DD_64(
/* |VR| */ void* QUAD_INTEGER_LENGTH ,
/* |VR|VL| */ void* DESCRIPTOR_PART ) ;

extern "C" int32_t LIB$SHOW_TIMER(
/* |VR| */ $DEFTYP* HANDLE_ADDRESS _Imm(0),
/* |VR| */ int32_t* CODE _Imm(0),
/* |PR|PROC|BO|AS| %IMM| */ uint32_t USER_ACTION_PROCEDURE =
0,
/* |PR| %IMM| */ uint32_t USER_ARGUMENT_VALUE = 0)
;

extern "C" int32_t LIB$SHOW_VM(
/* |VR| */ int32_t* CODE _Imm(0),
/* |PR|PROC|BO|AS| %IMM| */ uint32_t USER_ACTION_PROCEDURE =
0,
/* |PR| %IMM| */ uint32_t USER_SPECIFIED_ARGUMENT =
0) ;

extern "C" int32_t LIB$SHOW_VM_64(
/* |VR| */ void* CODE _Imm(0),
/* |PR|PROC|BO|AS| %IMM| */ uint32_t USER_ACTION_PROCEDURE =
0,
/* |PR| %IMM| */ int32_t USER_SPECIFIED_ARGUMENT =
0) ;

```

```

VAR floating_point_result : [VOLATILE] D_FLOAT$$TYPE) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$polyf (
    polynomial_argument : SINGLE;
    degree : $WORD;
    %REF coefficient : ARRAY [$l3..$u3:INTEGER] OF SINGLE;
    VAR floating_point_result : [VOLATILE] SINGLE) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$polyg (
    polynomial_argument : G_FLOAT$$TYPE;
    degree : $WORD;
    %REF coefficient : ARRAY [$l3..$u3:INTEGER] OF
G_FLOAT$$TYPE;
    VAR floating_point_result : [VOLATILE] G_FLOAT$$TYPE) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$polyh (
    polynomial_argument : QUADRUPLE;
    degree : $WORD;
    %REF coefficient : ARRAY [$l3..$u3:INTEGER] OF QUADRUPLE;
    VAR floating_point_result : [VOLATILE] QUADRUPLE) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$put_common (
    source_string : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER]
OF CHAR;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$put_invo_registers (
    %IMMED invo_handle : UNSIGNED;
    invo_context : $DEFPTR;
    invo_mask : $UQUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$put_output (
    message_string : [CLASS_S] PACKED ARRAY
[$l1..$u1:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$radix_point (
    VAR radix_point_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$l1..$u1:INTEGER] OF CHAR;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

```

```

extern "C" int32_t LIB$SHOW_VM_ZONE(
/* |VR|                               */ uint32_t* ZONE_ID ,
/* |VR|                               */ int32_t* DETAIL_LEVEL_Imm(0),
/* |PR|PROC|BO|AS|%IMM|              */ uint32_t USER_ACTION_PROCEDURE =
0,
/* |PR|%IMM|                          */ uint32_t USER_ARG = 0) ;

extern "C" int32_t LIB$SHOW_VM_ZONE_64(
/* |VR|                               */ void* ZONE_ID ,
/* |VR|                               */ void* DETAIL_LEVEL_Imm(0),
/* |PR|PROC|BO|AS|%IMM|              */ uint32_t USER_ACTION_PROCEDURE =
0,
/* |PR|%IMM|                          */ int32_t USER_ARG = 0) ;

extern "C" void LIB$SIGNAL(
/* |PR|%IMM|                          */ uint32_t CONDITION_VALUE ,
/* |PR|%IMM|                          */ int32_t NUMBER_OF_ARGUMENTS = 0,
/* |LS|                                */ int32_t _ListArgs = 0, ...)
;

extern "C" int32_t LIB$SIG_TO_RET(
/* |VR|US|%REF|                       */ void* SIGNAL_ARGUMENTS ,
/* |VR|US|%REF|                       */ void* MECHANISM_ARGUMENTS ) ;

extern "C" int32_t LIB$SIG_TO_STOP(
/* |VR|US|VL|%REF|                    */ void* SIGNAL_ARGUMENTS ,
/* |VR|US|%REF|                       */ void* MECHANISM_ARGUMENTS ) ;

extern "C" int32_t LIB$SIM_TRAP(
/* |VR|US|VL|%REF|                    */ void* SIGNAL_ARGUMENTS ,
/* |VR|US|%REF|                       */ void* MECHANISM_ARGUMENTS ) ;

extern "C" int32_t LIB$SKPC(
/* |VR| %C_S|                          */ DXX& CHARACTER_STRING ,
/* |VR| %C_S|                          */ DXX& SOURCE_STRING ) ;

extern "C" int32_t LIB$SPANC(
/* |VR| %C_S|                          */ DXX& SOURCE_STRING ,
/* |VR| %REF|                          */ void* TABLE_ARRAY ,
/* |VR|                                */ uchar_t* BYTE_INTEGER_MASK ) ;

extern "C" int32_t LIB$SPAWN(
/* |VR| %C_S|                          */ DXX& COMMAND_STRING_RefZero(),

```

```

[ASYNCHRONOUS] FUNCTION lib$remqhi (
    VAR header : [VOLATILE] $QUAD;
    VAR remque_address : [VOLATILE] $DEFPTR;
    retry_count : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$remqhiq (
    VAR header : [VOLATILE] $OCTA;
    VAR remque_address : [VOLATILE] lib$routines$$styp10;
    retry_count : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$remqti (
    VAR header : [VOLATILE] $QUAD;
    VAR remque_address : [VOLATILE] $DEFPTR;
    retry_count : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$remqtiq (
    VAR header : [VOLATILE] $OCTA;
    VAR remque_address : [VOLATILE] lib$routines$$styp11;
    retry_count : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$rename_file (
    old_filespec : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
    OF CHAR;
    new_filespec : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER]
    OF CHAR;
    default_filespec : [CLASS_S] PACKED ARRAY
[$13..$u3:INTEGER] OF CHAR := %IMMED 0;
    related_filespec : [CLASS_S] PACKED ARRAY
[$14..$u4:INTEGER] OF CHAR := %IMMED 0;
    flags : UNSIGNED := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_success_procedure := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_error_procedure := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_confirm_procedure := %IMMED 0;
    %IMMED user_specified_argument : UNSIGNED := %IMMED 0;
    VAR old_resultant_name : [CLASS_S,VOLATILE] PACKED ARRAY
[$110..$u10:INTEGER] OF CHAR := %IMMED 0;
    VAR new_resultant_name : [CLASS_S,VOLATILE] PACKED ARRAY
[$111..$u11:INTEGER] OF CHAR := %IMMED 0;
    VAR file_scan_context : [VOLATILE] UNSIGNED := %IMMED 0)
: INTEGER; EXTERNAL;

```

```

/* |VR| %C_S| */ DXX& INPUT_FILE _RefZero(),
/* |VR| %C_S| */ DXX& OUTPUT_FILE _RefZero(),
/* |VR| */ uint32_t* FLAGS _Imm(0),
/* |VR| %C_S| */ DXX& PROCESS_NAME _RefZero(),
/* |VR| VL| */ uint32_t* PROCESS_ID _Imm(0),
/* |PR| %IMM| */ $DEFTYP COMPLETION_STATUS_ADDRESS
= 0,
/* |VR| */ uchar_t*
BYTE_INTEGER_EVENT_FLAG_NUM _Imm(0),
/* |PR| PROC|BO|AS| %IMM| */ uint32_t AST_ADDRESS = 0,
/* |PR| US| %IMM| */ int32_t VARYING_AST_ARGUMENT = 0,
/* |VR| %C_S| */ DXX& PROMPT_STRING _RefZero(),
/* |VR| %C_S| */ DXX& CLI _RefZero(),
/* |VR| %C_S| */ DXX& TABLE _RefZero() ;

extern "C" int32_t LIB$STAT_TIMER(
/* |VR| */ int32_t* CODE ,
/* |VR| US|VL| %REF| */ void* VALUE_ARGUMENT ,
/* |VR| */ $DEFTYP* HANDLE_ADDRESS _Imm(0)) ;

extern "C" int32_t LIB$STAT_VM(
/* |VR| */ int32_t* CODE ,
/* |VR| US|VL| %REF| */ void* VALUE_ARGUMENT ) ;

extern "C" int32_t LIB$STAT_VM_64(
/* |VR| */ void* CODE ,
/* |VR| US|VL| %REF| */ void* VALUE_ARGUMENT ) ;

extern "C" void LIB$STOP(
/* |PR| %IMM| */ uint32_t CONDITION_VALUE ,
/* |PR| %IMM| */ int32_t NUMBER_OF_ARGUMENTS = 0,
/* |LS| */ int32_t _ListArgs = 0, ...)
;

extern "C" int32_t LIB$SUBX(
/* |VR| US| %REF| */ void* MINUEND_ARRAY ,
/* |VR| US| %REF| */ void* SUBTRAHEND_ARRAY ,
/* |VR| US|VL| %REF| */ void* DIFFERENCE_ARRAY ,
/* |VR| */ int32_t* ARRAY_LENGTH _Imm(0)) ;

extern "C" int32_t LIB$SUB_TIMES(
/* |VR| */ void* TIME1 ,
/* |VR| */ void* TIME2 ,
/* |VR| VL| */ void* RESULTANT_TIME ) ;

```

```

[ASYNCHRONOUS] FUNCTION lib$reserve_ef (
    event_flag_number : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$reset_vm_zone (
    zone_id : UNSIGNED) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$reset_vm_zone_64 (
    zone_id : $UQUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$run_program (
    program_name : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
    OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$scanc (
    source_string : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
    OF CHAR;
    %REF table_array : ARRAY [$12..$u2:INTEGER] OF $UBYTE;
    byte_integer_mask : $UBYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$scopy_dxdx (
    source_string : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
    OF CHAR;
    VAR destination_string : [CLASS_S,VOLATILE] PACKED ARRAY
    [$12..$u2:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$scopy_r_dx (
    word_integer_source_length : $UWORD;
    %REF source_string : PACKED ARRAY [$12..$u2:INTEGER] OF
    CHAR;
    VAR destination_string : [CLASS_S,VOLATILE] PACKED ARRAY
    [$13..$u3:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$scopy_r_dx_64 (
    quad_integer_source_length : $UQUAD;
    %REF source_string : PACKED ARRAY [$12..$u2:INTEGER] OF
    CHAR;
    VAR destination_string : [CLASS_S,VOLATILE] PACKED ARRAY
    [$13..$u3:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$set_logical (
    logical_name : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
    OF CHAR;
    value_string : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER]
    OF CHAR := %IMMED 0;

```

```

extern "C" int32_t LIB$SYS_ASCTIM(
/* |VR|VL| */ uint16_t* RESULTANT_LENGTH _Imm(0),
/* |VR|VL| %C_S| */ DXX& TIME_STRING_RefZero(),
/* |VR| */ void* USER_TIME _Imm(0),
/* |VR| */ uint32_t* FLAGS _Imm(0) );

extern "C" int32_t LIB$SYS_FAO(
/* |VR| %C_S| */ DXX& CHARACTER_STRING ,
/* |VR|VL| */ uint16_t* RESULTANT_LENGTH _Imm(0),
/* |VR|VL| %C_S| */ DXX& RESULTANT_STRING_RefZero(),
/* |LS| */ int32_t _ListArgs = 0, ...)
;

extern "C" int32_t LIB$SYS_FAO_L(
/* |VR| %C_S| */ DXX& CHARACTER_STRING ,
/* |VR|VL| */ uint16_t* RESULTANT_LENGTH _Imm(0),
/* |VR|VL| %C_S| */ DXX& RESULTANT_STRING_RefZero(),
/* |VR| */ uint32_t*
DIRECTIVE_ARGUMENT_ADDRESS _Imm(0) );

extern "C" int32_t LIB$SYS_FAO_L_64(
/* |VR| %C_S| */ DXX& CHARACTER_STRING ,
/* |VR|VL| */ uint16_t* RESULTANT_LENGTH _Imm(0),
/* |VR|VL| %C_S| */ DXX& RESULTANT_STRING_RefZero(),
/* |VR| */ void* DIRECTIVE_ARGUMENT_ADDRESS
_Imm(0) );

extern "C" int32_t LIB$SYS_GETMSG(
/* |VR| */ uint32_t* MESSAGE_ID ,
/* |VR|VL| */ uint16_t* MESSAGE_LENGTH _Imm(0),
/* |VR|VL| %C_S| */ DXX& DESTINATION_STRING_RefZero(),
/* |VR| */ uint32_t* FLAGS _Imm(0),
/* |VR|US|VL| %REF| */ void* UNSIGNED_RESULTANT_ARRAY
_Imm(0) );

extern "C" int32_t LIB$SYS_TRNLOG(
/* |VR| %C_S| */ DXX& LOGICAL_NAME ,
/* |VR|VL| */ uint16_t* WORD_INTEGER_DEST_LENGTH
_Imm(0),
/* |VR|VL| %C_S| */ DXX& DESTINATION_STRING_RefZero(),
/* |VR|VL| */ char* BYTE_INTEGER_TABLE _Imm(0),
/* |VR|VL| */ uchar_t* ACCESS_MODE _Imm(0),
/* |VR| */ uchar_t* BYTE_INTEGER_DISABLE_MASK
_Imm(0) );

extern "C" int32_t LIB$TABLE_PARSE(

```

```

table : [CLASS_S] PACKED ARRAY [$13..$u3:INTEGER] OF CHAR
:= %IMMED 0;
attributes : UNSIGNED := %IMMED 0;
%REF item_list : [UNSAFE] ARRAY [$15..$u5:INTEGER] OF
$UBYTE := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$set_symbol (
symbol : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
CHAR;
value_string : [CLASS_S] PACKED ARRAY [$12..$u2:INTEGER]
OF CHAR;
table_type_indicator : INTEGER := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sfreel_dd (
VAR descriptor_address : [VOLATILE] $UQUAD) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sfreen_dd (
number_of_descriptors : UNSIGNED;
%REF first_descriptor_array : [VOLATILE] ARRAY
[$12..$u2:INTEGER] OF $UQUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$ssetl_dd (
word_integer_length : $UWORD;
VAR descriptor_part : [VOLATILE] $UQUAD) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$ssetl_dd_64 (
quad_integer_length : $UQUAD;
VAR descriptor_part : [VOLATILE] $UQUAD) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$show_timer (
handle_address : $DEFPTR := %IMMED 0;
code : INTEGER := %IMMED 0;
%IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_action_procedure := %IMMED 0;
%IMMED user_argument_value : UNSIGNED := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$show_vm (
code : INTEGER := %IMMED 0;
%IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_action_procedure := %IMMED 0;
/* |VR|US|VL|%REF| */ void* ARGUMENT_BLOCK ,
/* |VR|US|%REF| */ void* STATE_TABLE ,
/* |VR|US|%REF| */ void* KEY_TABLE ) ;

extern "C" int32_t LIB$TRAVERSE_TREE(
/* |VR| */ /* $DEFTYP* TREEHEAD ,
/* |PR|PROC|BO|AS|%IMM| */ uint32_t USER_ACTION_PROCEDURE ,
/* |VR|US|%REF| */ void* USER_DATA_ADDRESS _Imm(0) ) ;

extern "C" int32_t LIB$TRAVERSE_TREE_64(
/* |VR| */ /* void* TREEHEAD ,
/* |PR|PROC|BO|AS|%IMM| */ uint32_t USER_ACTION_PROCEDURE ,
/* |VR|US|%REF| */ void* USER_DATA_ADDRESS _Imm(0) ) ;

extern "C" int32_t LIB$TRA_ASC_EBC(
/* |VR|%C_S| */ /* DXX& SOURCE_STRING ,
/* |VR|VL|%C_S| */ /* DXX& BYTE_INTEGER_DEST_STRING ) ;

extern "C" int32_t LIB$TRA_EBC_ASC(
/* |VR|%C_S| */ /* DXX& BYTE_INTEGER_SOURCE_STRING ,
/* |VR|VL|%C_S| */ /* DXX& DESTINATION_STRING ) ;

extern "C" int32_t LIB$TRIM_FILESPEC(
/* |VR|%C_S| */ /* DXX& OLD_FILESPEC ,
/* |VR|VL|%C_S| */ /* DXX& NEW_FILESPEC ,
/* |VR| */ /* uint16_t* WORD_INTEGER_WIDTH
_Imm(0),
/* |VR|VL| */ /* uint16_t* RESULTANT_LENGTH _Imm(0)
;

extern "C" int32_t LIB$TRIM_FULLNAME(
/* |VR|%C_S| */ /* DXX& FULLNAME ,
/* |VR|VL|%C_S| */ /* DXX& TRIMMED_NODENAME ,
/* |VR| */ /* uint16_t* OUTPUT_WIDTH _Imm(0),
/* |VR|VL| */ /* uint16_t* RESULTANT_LENGTH _Imm(0)
;

extern "C" int32_t LIB$UID_TO_ASCII(
/* |VR| */ /* void* INPUT_UID ,
/* |VR|VL|%C_S| */ /* DXX& OUTPUT_STRING ) ;

extern "C" int32_t LIB$VERIFY_VM_ZONE(
/* |VR| */ /* uint32_t* ZONE_ID ) ;

```

```

    %IMMED user_specified_argument : UNSIGNED := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$show_vm_64 (
    code : $QUAD := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_action_procedure := %IMMED 0;
    %IMMED user_specified_argument : INTEGER := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$show_vm_zone (
    zone_id : UNSIGNED;
    detail_level : INTEGER := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_action_procedure := %IMMED 0;
    %IMMED user_arg : UNSIGNED := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$show_vm_zone_64 (
    zone_id : $UQUAD;
    detail_level : $QUAD := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_action_procedure := %IMMED 0;
    %IMMED user_arg : INTEGER := %IMMED 0) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] PROCEDURE lib$signal (
    %IMMED condition_value : UNSIGNED;
    %IMMED number_of_arguments : INTEGER := %IMMED 0;
    %IMMED FAO_argument : [LIST,UNSAFE] INTEGER); EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sig_to_ret (
    %REF signal_arguments : [UNSAFE] ARRAY [$l1..$u1:INTEGER]
OF $UBYTE;
    %REF mechanism_arguments : [UNSAFE] ARRAY
[$l2..$u2:INTEGER] OF $UBYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sig_to_stop (
    %REF signal_arguments : [VOLATILE,UNSAFE] ARRAY
[$l1..$u1:INTEGER] OF $UBYTE;
    %REF mechanism_arguments : [UNSAFE] ARRAY
[$l2..$u2:INTEGER] OF $UBYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sim trap (

```

```

extern "C" int32_t LIB$VERIFY_VM_ZONE_64(
/* |VR|                                     */ void* ZONE_ID ) ;

extern "C" int32_t LIB$WAIT(
/* |VR|                                     */ float* SECONDS ,
/* |VR|                                     */ uint32_t* FLAGS _Imm(0),
/* |VR|                                     */ uint32_t* FLOAT_TYPE _Imm(0)) ;
void _QA_FUNCTION_PROTOTYPES_init(void)
{
    static bool _was_initialized = false;
    if (!_was_initialized)
        return;
    _was_initialized = true;
}
/* MODULE SELF CHECK END */

class QA_FUNCTION_PROTOTYPESToDo : public ToDo {
public:
    QA_FUNCTION_PROTOTYPESToDo() :
ToDo("QA_FUNCTION_PROTOTYPES") { OnBegin(); };
    ~QA_FUNCTION_PROTOTYPESToDo() { OnEnd(); };
    void OnBegin();
    void OnEnd();
};
static QA_FUNCTION_PROTOTYPESToDo todo;
void QA_FUNCTION_PROTOTYPESToDo::OnBegin() {
    _QA_FUNCTION_PROTOTYPES_init();
}
void QA_FUNCTION_PROTOTYPESToDo::OnEnd() {
}

```

```

%REF signal_arguments : [VOLATILE,UNSAFE] ARRAY
[$l1..$u1:INTEGER] OF $BYTE;
%REF mechanism_arguments : [UNSAFE] ARRAY
[$l2..$u2:INTEGER] OF $BYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$skpc (
    character_string : [CLASS_S] PACKED ARRAY
[$l1..$u1:INTEGER] OF CHAR;
    source_string : [CLASS_S] PACKED ARRAY [$l2..$u2:INTEGER]
OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$spanc (
    source_string : [CLASS_S] PACKED ARRAY [$l1..$u1:INTEGER]
OF CHAR;
    %REF table_array : ARRAY [$l2..$u2:INTEGER] OF $BYTE;
    byte_integer_mask : $BYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$spawn (
    command_string : [CLASS_S] PACKED ARRAY
[$l1..$u1:INTEGER] OF CHAR := %IMMED 0;
    input_file : [CLASS_S] PACKED ARRAY [$l2..$u2:INTEGER] OF
CHAR := %IMMED 0;
    output_file : [CLASS_S] PACKED ARRAY [$l3..$u3:INTEGER]
OF CHAR := %IMMED 0;
    flags : UNSIGNED := %IMMED 0;
    process_name : [CLASS_S] PACKED ARRAY [$l5..$u5:INTEGER]
OF CHAR := %IMMED 0;
    VAR process_id : [VOLATILE] UNSIGNED := %IMMED 0;
    %IMMED completion_status_address : $DEFPTR := %IMMED 0;
    byte_integer_event_flag_num : $BYTE := %IMMED 0;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE AST_address :=
%IMMED 0;
    %IMMED varying_AST_argument : [UNSAFE] INTEGER := %IMMED
0;
    prompt_string : [CLASS_S] PACKED ARRAY
[$l11..$u11:INTEGER] OF CHAR := %IMMED 0;
    cli : [CLASS_S] PACKED ARRAY [$l12..$u12:INTEGER] OF CHAR
:= %IMMED 0;
    table : [CLASS_S] PACKED ARRAY [$l13..$u13:INTEGER] OF
CHAR := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$stat_timer (
    code : INTEGER;
    %REF value_argument : [VOLATILE,UNSAFE] ARRAY
[$l2..$u2:INTEGER] OF $BYTE;
    handle_address : $DEFPTR := %IMMED 0) : INTEGER;
EXTERNAL;

```

```

[ASYNCHRONOUS] FUNCTION lib$stat_vm (
    code : INTEGER;
    %REF value_argument : [VOLATILE,UNSAFE] ARRAY
[$12..$u2:INTEGER] OF $UBYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$stat_vm_64 (
    code : $QUAD;
    %REF value_argument : [VOLATILE,UNSAFE] ARRAY
[$12..$u2:INTEGER] OF $UBYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] PROCEDURE lib$stop (
    %IMMED condition_value : UNSIGNED;
    %IMMED number_of_arguments : INTEGER := %IMMED 0;
    %IMMED FAO_argument : [LIST,UNSAFE] INTEGER); EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$subx (
    %REF minuend_array : [UNSAFE] ARRAY [$11..$u1:INTEGER] OF
$UBYTE;
    %REF subtrahend_array : [UNSAFE] ARRAY [$12..$u2:INTEGER]
OF $UBYTE;
    %REF difference_array : [VOLATILE,UNSAFE] ARRAY
[$13..$u3:INTEGER] OF $UBYTE;
    array_length : INTEGER := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sub_times (
    time1 : $UQUAD;
    time2 : $UQUAD;
    VAR resultant_time : [VOLATILE] $UQUAD) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sys_asctim (
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0;
    VAR time_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR;
    user_time : $UQUAD := %IMMED 0;
    flags : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sys_fao (
    character_string : [CLASS_S] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0;
    VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$13..$u3:INTEGER] OF CHAR;

```



```

%REF directive_argument : [LIST,UNSAFE] ARRAY
[$14..$u4:INTEGER] OF $BYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sys_faol (
  character_string : [CLASS_S] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
  VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0;
  VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$13..$u3:INTEGER] OF CHAR;
  directive_argument_address : UNSIGNED) : INTEGER;
EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sys_faol_64 (
  character_string : [CLASS_S] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
  VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0;
  VAR resultant_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$13..$u3:INTEGER] OF CHAR;
  directive_argument_address : $UQUAD) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sys_getmsg (
  message_id : UNSIGNED;
  VAR message_length : [VOLATILE] $UWORD := %IMMED 0;
  VAR destination_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$13..$u3:INTEGER] OF CHAR;
  flags : UNSIGNED := %IMMED 0;
  %REF unsigned_resultant_array : [VOLATILE,UNSAFE] ARRAY
[$15..$u5:INTEGER] OF $BYTE := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$sys_trnlog (
  logical_name : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
OF CHAR;
  VAR word_integer_dest_length : [VOLATILE] $UWORD :=
%IMMED 0;
  VAR destination_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$13..$u3:INTEGER] OF CHAR;
  VAR byte_integer_table : [VOLATILE] $BYTE := %IMMED 0;
  VAR access_mode : [VOLATILE] $BYTE := %IMMED 0;
  byte_integer_disable_mask : $BYTE := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$stable_parse (
  %REF argument_block : [VOLATILE,UNSAFE] ARRAY
[$11..$u1:INTEGER] OF $BYTE;
  %REF state_table : [UNSAFE] ARRAY [$12..$u2:INTEGER] OF
$BYTE;

```

```

%REF key_table : [UNSAFE] ARRAY [$13..$u3:INTEGER] OF
$SUBYTE) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$traverse_tree (
    treehead : $DEFPTR;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_action_procedure;
    %REF user_data_address : [UNSAFE] ARRAY
[$13..$u3:INTEGER] OF $SUBYTE := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$traverse_tree_64 (
    treehead : lib$routines$$styl2;
    %IMMED [UNBOUND, ASYNCHRONOUS] PROCEDURE
user_action_procedure;
    %REF user_data_address : [UNSAFE] ARRAY
[$13..$u3:INTEGER] OF $SUBYTE := %IMMED 0) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$tra_asc_ebc (
    source_string : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
OF CHAR;
    VAR byte_integer_dest_string : [CLASS_S,VOLATILE] PACKED
ARRAY [$12..$u2:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$tra_ebc_asc (
    byte_integer_source_string : [CLASS_S] PACKED ARRAY
[$11..$u1:INTEGER] OF CHAR;
    VAR destination_string : [CLASS_S,VOLATILE] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR) : INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$trim_filespec (
    old_filespec : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER]
OF CHAR;
    VAR new_filespec : [CLASS_S,VOLATILE] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR;
    word_integer_width : $UWORD := %IMMED 0;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

[ASYNCHRONOUS] FUNCTION lib$trim_fullname (
    fullname : [CLASS_S] PACKED ARRAY [$11..$u1:INTEGER] OF
CHAR;
    VAR trimmed_nodename : [CLASS_S,VOLATILE] PACKED ARRAY
[$12..$u2:INTEGER] OF CHAR;
    output_width : $UWORD := %IMMED 0;
    VAR resultant_length : [VOLATILE] $UWORD := %IMMED 0) :
INTEGER; EXTERNAL;

```

```
[ASYNCHRONOUS] FUNCTION lib$uid_to_ascii (  
    input_uid : $UOCTA;  
    VAR output_string : [CLASS_S,VOLATILE] PACKED ARRAY  
    [$12..$u2:INTEGER] OF CHAR) : INTEGER; EXTERNAL;  
  
[ASYNCHRONOUS] FUNCTION lib$verify_vm_zone (  
    zone_id : UNSIGNED) : INTEGER; EXTERNAL;  
  
[ASYNCHRONOUS] FUNCTION lib$verify_vm_zone_64 (  
    zone_id : $UQUAD) : INTEGER; EXTERNAL;  
  
[ASYNCHRONOUS] FUNCTION lib$wait (  
    seconds : SINGLE;  
    flags : UNSIGNED := %IMMED 0;  
    float_type : UNSIGNED := %IMMED 0) : INTEGER; EXTERNAL;  
  
END.
```