DADF Facility

US Detailed Design Specification - ICD: MUBM::Workstation

Document Signature Table

	Name	Function	Signature	Date
Author	Oliver Harrmann	US Element Manager		20.08.99
Approval				
Approval				
Release				
Eumetsat Approval				

Document Change Record

Issue/Revision	Date	DCN No.	Changed Pages/Paragraphs
1.0	02.07.98		Initial Release
2.0	07.12.98		Change bars mark changes. Due to a design change, all sections regarding the PN generator were deleted.
3.0	13.07.99		Change bars mark changes.

Distribution List

Internal Distribution	
Name	**No. Copies**
VCS Archive, Dev	2

External Distribution		
Company	**Name**	**No. Copies**
Eumetsat		4

Table of Contents

List of Figures

None.

List of Tables

1 Introduction

1.1 Purpose And Scope of the Interface Control Document

This document shall define the interface between the MSG User Station Baseband Module and the User Station Workstation or PC. The specification covers both a HRUS MUBM and a LRUS MUBM due to the fact, that the data streams are logically identical. The only difference is in performance. If necessary, this aspect will be mentioned separately for both systems throughout the document.

For this purpose

- the data items,
- the command set,
- how to use the interface and
- exception handling procedures

are defined so that the interface is logically specified for both sides, the software, running on a PC and the firmware, running on the MUBM system. In addition

- the physical aspects and
- the reference standard

are specified.

1.2 Document Structure

The structure of the document is as follows:

Chapter	Description
Introduction	This chapter gives an introduction of the document.
Interfaced Systems	This chapter describes the principle functions of both interfaced systems
Data Streams	Here all data streams, data structures and items are specified.
Protocol	Here the data protocol is specified. This comprises the coding of the data streams into the protocol, the command set, how to use the protocol and exception handling.
Hardware Interface	Here the hardware aspects of the interface are specified, the standard, the connectors, the electrical specification

1.3 Purpose of the Interface

This interface covers two main data streams within the User Station.

- The VCDU data stream including attached quality information according to the LRIT/HRIT data protocol
- The data stream to and from the SKU. Here, the MUBM only interacts as an interface converter. No data formatting is performed on the data to and from the SKU [FICD/102].

Additionally different monitor and control data streams are transferred via this interface.

1.4 Relationship to Other Interfaces

The MUBM::US Workstation interface also comprises a data stream dedicated to the SKU. Here the same data are transferred via this interface as via the SKU interface. This interface is specified in [FICD/102].

1.5 Open Issues And Assumptions

There are no open issues or assumptions made in this document.

1.6 Applicable And Reference Documents

1.6.1 Applicable Documents

Acronym	Title	Reference
FGLO	VCS DADF Glossary	
FICD/102	DADF ICD: SKU	EUM/MSG/ICD/102
FRS	MSG GS DADF Requirement Specification	EUM/MSG/SPE/041
MSI	HRIT/LRIT Mission Specific Implementation	EUM/MSG/SPE/057
SCSI	Small Computer System Interface-2	ANSI X3.141-1994
SGICD	Satellites to Ground Segment Interface Control Document	EUM/MSG/ICD/006
USDDS	DADF US Detailed Design Specification	EUM/MSG/SPE/176
USDJ	DADF User Stations Design Justification	EUM/MSG/SPE/128
USDS	DADF User Stations Design Specification	EUM/MSG/SPE/127

1.6.2 Reference Documents

Acronym	Title	Reference
LRIT\HRIT\GSP	LRIT/HRIT Global Specification	VCS/DADF/GEN/RESP/009

1.7 Conventions

For the definition of data items and structures the ANSI C like notation is used:

uint32	unsigned 32 bit integer value
uint16	unsigned 32 bit integer value
uchar	unsigned 8 bit integer value

int32	32 bit integer value
int16	32 bit integer value
char	8 bit integer value or character

The bytes are numbered according to [MSI], i.e. CCSDS convention. The bit numbering is used conventionally like bit 7 = msb, bit 0 = lsb. This complies with [SCSI]. Integer values of more than one byte are according to CCSDS and MSI convention.

In special cases a detailed description is provided at the corresponding location.

2 Interfaced Systems

The interfaced systems are

- the MSG User Station Baseband Module (either for LRUS or for HRUS) and
- the User Station Workstation.

According to the used interface standard [SCSI], the US Workstation works as a SCSI initiator, the MUBM as SCSI target system. Herewith it is clear, that data transfers are always initiated from the US Workstation side.

The main function of the MUBM is to receive data from the MSG spacecraft, to process these data to data packets, the so-called Virtual Channel Data Units (VCDU), and to provide the buffered data packets via the MUBM::US Workstation interface to the US Workstation. It shall be noted that the VCDUs are transferred together the frame synchronisation marker and the Reed-Solomon check symbols, which forms the decoded CVCDU, and, furthermore, quality information.

On the other hand, the US Workstation software has to retrieve the VCDU data packets via the interface for further processing according to the LRIT/HRIT protocol described in the [MSI]. It shall be noted that according to the limited buffer size within the MUBM, the host PC, i.e. the US Workstation, has to retrieve the data packets with a certain average data rate according to the transmission rates of the HRIT or the LRIT channel. The data rates are specified in section 3.

A second functionality of the MUBM is to provide the interface to the Station Key Unit SKU [FICD/102]. In order to calculate the PNK, the appropriate data are transferred from the Host to the MUBM. After the SKU has calculated the PNK, the data can be retrieved for further processing. Another function is supported, the storage of public keys. Here the PBKs are transferred from the Host via the interface to the MUBM and from here to the SKU itself.

For Monitoring and control purposes, additional data flows are specified. Depending on the individual data items, both systems, the MUBM and the US Workstation have to perform different functions.

3 Data Streams Overview

3.1 Summary

This section specifies all data streams and data items used within the SCSI protocol.

3.2 Data Definitions

The following table contains all data definitions used within this ICD including the value ranges of the different data items. The data items are ordered according to the data streams.

Data Name	Data Type	Data Description
XritSyncMarker[4]	Uchar	Synchronisation marker, refer to [MSI]
XritVCDU_data[892]	Uchar	VCDU data, refer to [MSI]
XritVCDU_RS[128]	Uchar	Reed Solomon check symbols, refer to [MSI]
XritVCDU_qual[208] {	Uchar	This data array comprises a packed structure of quality information, which is described in detail below.
T_bb_qual[8] {	Structure	Array of 8 times the following structure. The quality information is synchronised to the VCDU/CVCDU above. The synchronisation error is max ±⅛ CVCDU for the Viterbi results. Please note that every element of this structure is doubled due to the fact that a Viterbi measurement is only possible every 4000 symbols. Due to this the symbol error rate is $\frac{1}{8 \cdot 4000} \sum_{8} ViterbiDecBER$
InpSigOffsFreq	Int32	NCO value of demodulator PLL offset frequency: $f_{c,offset}$ = InputSignalOffsetFrequency * 0.01 Hz
BitrateOffsFreq	Int32	NCO value of bit synchroniser PLL offset frequency: $f_{b,offset}$ = BitrateOffsetFrequency * 0.01 Hz
ViterbiDecBER	Uint16	Measured symbol error rate per 4000 symbols; due to the function of the Viterbi Decoder IC [Q1900]
BsFlags	uint16	Demodulator bitsynchroniser and Viterbi flags. Bit map of following flags (error/not locked if bit is set)
	bit0	- Demodulator lock
	bit1	- Bit synchroniser lock
	bit2	- Input level too low
	bit3	- Input level too high

Data Name	Data Type	Data Description
	bit4 bit5-bit16	- Viterbi node sync lock - Reserved for future use
InpSigAGC	Unit32	Hex value of the input AGC on the I960-2nd DC, controlled by the basebandProc: $V_{Amplifier}$ = inpSigAGC * 0.32 dB InpSigAGC range: 127 …256 (delta = 40 dB) InpSigAGC max=255: Signal lost InpSigAGC max=120: Signal level overflow
BaseSigAGC	Unit32	BASEBAND SIGNAL AGC INSIDE THE HARRIS [HSP50214].
Dummy[4]	Uchar	Dummy bytes to fill up the structure to 4 byte boundary
}		
BitErr[4]	Uchar	4 bit error counters indicating the number of corrected bits (one counter per Reed-Solomon block) range: 0 … 128 set to 0 when RS fails
ByteErr[4]	Uchar	4 byte error counters indicating the number of corrected bytes (one counter per Reed-Solomon block) range: 0 … 16 set to when RS fails
TimeStamp	Unit32	MUBM timestamp (2 ms modulo counter) at baseband processing stage
SyncErrs	Uchar	Bit errors in synchronisation marker 0x00 to maximum allowed bit errors. Sync pattern was found in datastream and value represents number of bit errors in sync pattern. A bigger value determines sync pattern not found.
BitSlip	Char	Sync misalignment to expected location invalid if no sync found (refer to VCDUFlags)
VCDUFlags	uint16 bit0 bit1 bit2 bit3	Bit map of following flags (error occurred if bit is set) - RS error flag 0 - RS error flag 1 - RS error flag 2 - RS error flag 3 RS error flag is set with an uncorrectable error

Data Name	Data Type	Data Description
	bit4 . . bit5 – bit13 bit14 bit15	- frame sync flag (set when out of lock) Note: If frame sync not locked, RS decoding is not performed. Accordingly the corresponding quality information is not valid. Reserved for future use FS_NOTFOUND flag: Set if no sync was found but data is processed due to configured fly-wheel operation. FS_INV flag: Set if data is inverted due to an inverted sync pattern
}		
SkuCommandData[]	Uchar	This character array contains the SKU command data, specified in the SKU ICD [FICD/102]
SkuReturnData[]	Uchar	This character array contains the SKU return data, specified in the SKU ICD [FICD/102]
CtrlFsyncMaxNoBitSlips	Uchar	Control parameter defining the allowed number of bit (±)slips of the frame synchroniser Range: 0 – 5 Default: 3
CtrlFsyncMaxNoBitErrors	Uchar	Control parameter defining the allowed number of bit errors in the frame synchronisation marker Range: 0 – 8 Default: 3
CtrlFsyncPatternLength	Uchar	Control parameter defining the length of the synchronisation marker: Default: 32
CtrlFsyncUnlockThreshold	Uchar	Control parameter defining the fly wheel parameter of the frame synchroniser (number of unfound fsync markers before frame sync lock loss is detected): Range: 1 – 20 Default: 3
CtrlFsyncLockThreshold	Uchar	Control parameter defining the 2nd fly wheel parameter of the frame synchroniser (number of found fsync markers before frame sync lock is detected): Range: 1 – 20

Data Name	Data Type	Data Description
		Default: 3
Ctrldummy[27]	Uchar	Undefined data for future use.
		Monitoring data The monitoring data mon* are calculated from the corresponding values attached to the xritVCDU data. These data are averaged, accumulated or traced over one satellite revolution, which yields to 0.6 sec. As the evaluation of monitoring parameter is linked to the dissemination of VCDUs, the following parameter are calculated over 11 VCDUs for LRIT and 85 VCDUs for HRIT.
MonTimeStamp	Uint32	MUBM 2 msec timer, modulo 2^{32} bit counter; actual timer value is actualised before the monitoring data are transferred via the SCSI bus
MonInpSigOffsFreq	int32	Average NCO value of demodulator PLL offset frequency: $f_{c,offset}$ = InpSigOffsFreq * 0.01 Hz
MonBitrateOffsFreq	int32	Average NCO value of bit synchroniser PLL offset frequency: $f_{c,offset}$ = BitrateOffsFreq * 0.01 Hz
MonViterbiDecBER	Uint32	Average measured symbol error rate per 11 or 85 times 16000 symbols; due to the function of the Viterbi Decoder IC [Q1900]
MonInpSigAGC	Uint32	Average hex value of the input AGC on the I960-2nd DC, controlled by the basebandProc: $V_{Amplifier}$ = inpSigAGC * 0.32 dB InpSigAGC range: 120 …256 (delta = 40 dB) InpSigAGC max=255: Signal lost InpSigAGC max=120: Signal level overflow
MonBaseSigAGC	Uint32	AVERAGE BASEBAND SIGNAL AGC INSIDE THE HARRIS [HSP50214]
FreeSpaceBpBitstreamBuf;	Uint32	Defines the free space of the bitstream buffer on the baseband module in number of elements of the BitstreamBuffer on the baseband processing module. Each element contains 4 KB of bitstream data plus the corresponding quality information. Maximum value 208
MonBSFlags	Uint16	A detailed description of the single bits is given in BSFlags. Whenever a bit was set in the BSFlags during the monitoring time, the corresponding bit is

Data Name	Data Type	Data Description
		set in monBSFlags
MonEbN0	Char	Eb/No calculated on base of Viterbi bit error rate; please note that this value is an estimation based on short term measurements.
MonBPdummy[357]	Uchar	Filler characters.
MonCorrBitErr	Uint32	Number of RS corrected bit errors
MonCorrByteErr	Uint32	Number of RS corrected byte errors
MonxStPfl	Uint32	Number of lost VCDUs due to frame sync loss or RS failure; cyclic counter which is not reset at the begin of a measurement
MonBitSlip	Char	Maximum sync misalignment to expected location during measurement period. Invalid if no sync found (refer to VCDUFlags)
MonFsyncErr	Uchar	Number of bit errors in frame sync marker over measurement period
MonVCDUFlags	Uint16	A detailed description of the single bits is given in VCDUFlags. Whenever a bit was set in the VCDUFlags during the monitoring time, the corresponding bit is set in monVCDUFlags
FreeSpaceXdBitstreamBuf	Uint32	Defines the free space of the bitstream buffer on the Xrit decoder. The value is given in numbers of elements of the BitstreamBuffer. Each element contains 4 KB of bitstream data plus the corresponding quality information. Maximum value 104
FreeSpaceXdVcduBuf	Uint32	Defines the free space of the VCDU buffer on the Xrit decoder. The value is given in numbers of elements of the VCDU Buffer. Each element contains one XritVcdu structure. Maximum value 412
MonXDdummy[360]	Uchar	Filler characters
FreeSpaceMcLogMessageBuf	Uint32	Defines the free space of the log message buffer on the MUBM control module in number of log messages. Maximum value: 512
AllocSpaceMcLogMessageBuf	Uint32	Defines the number log messages in the log message buffer on the MUBM control module. Maximum value: 512
MonMCdummy[244]	Uchar	Filler characters
		Log messages of the MUBM are composed of the following character fields:
LogSeverity[2]	Char	'E ' error

Data Name	Data Type	Data Description
		'W ' warning 'I ' information 'T ' trace
ModuleId[4]	Char	'BPR ' basebandProc module 'XRD ' xritDec module 'MCB ' mubmCtr_basebandProcTask 'MCX ' mubmCtr_xritDecTask 'MCP ' mubmCtr_pnGenTask 'MCK ' mubmCtr_skuTask 'MCS ' mubmCtr_scsiTask
MubmTimer[9]	Char	"<ASCII coded 2ms timer value, 8 digits>' '
Message[49]	Char	"<ASCII message text, 48 characters followed by <LF>"

Table 1: Data Definition

3.3 Data streams

3.3.1 VCDU Data Stream

Description

The following structure defines one packet of received data in the MUBM which are forwarded to the US Workstation. According to the [MSI] a continuous stream of these packets is expected. In the case, that the dissemination system DISE does not provide data, filler VCDUs are generated at PGS according to the [MSI]. A detailed definition of the attached VCDU quality information is given in section 3.2. It shall be noted that even in the case, the MUBM does not receive a valid signal, synchronisers are not locked etc., the VCDU data stream is generated to provide the quality information but without real data.

Data Flow Direction

The data flow is from MUBM to US Workstation.

Data Type and Size

```
struct US_XritVcdu {
        uchar   xritSyncMarker[4]       // synchronisation marker, refer to [MSI]
        uchar   xritVCDU_data[892]      // VCDU data, refer to [MSI]
        uchar   xritVCDU_RS[128]        // Reed Solomon check symbols, refer to [MSI]
        uchar   xritVCDU_qual[208]      // refer to data definition for detailed description
};
```

Data Rate and Frequency

According to the [SGICD], the average data rate of pure VCDU data is for

HRIT 1 Mbps or 125.000 Bytes per sec,

LRIT 128 Kbps or 16.000 Bytes per sec.

Taking into account the overhead caused by the above-mentioned data, the data rate yield to

HRIT 163.677 Bytes per sec ~~[TBC]~~

LRIT 20.951 Bytes per sec. ~~[TBC]~~

3.3.2 SKU_command_string Data Stream

Description

The string is used to forward it to the SKU in order to allow the performance of all SKU commands.

Data Flow Direction

The data flow is from US Workstation to MUBM and then to the SKU (controlled by the MUBM).

Data Type And Size

```
struct SKU_command_string {
        uchar   skuCommandData[]            // refer to SKU ICD [FICD/102]
};
```

Data Rate And Frequency

This cannot be specified. In principle the transfer of SKU command strings is necessary to allow the PNK calculation, which is to be don once every encrypted file. The file rate is not constant but could be estimated to about 1.4 per second average.

3.3.3 SKU_return_string Data Stream

Description

The string is used to forward the SKU return data to the workstation.

Data Flow Direction

The data flow is from MUBM to US Workstation.

Data Type and Size

```
struct US_SkuPnk {
        uchar   skuReturnData[]             //  refer to SKU ICD [FICD/102]
};
```

Data Rate and Frequency

This cannot be specified. In principle the transfer of SKU return strings is necessary to allow the PNK calculation, which is to be don once every encrypted file. The file rate is not constant but could be estimated to about 1.4 per second average.

3.3.4 Control Baseband Process Data Stream

Description

The following structure defines the data structure of parameter, which are used to control the baseband processing of the MUBM, especially the frame synchroniser.

Data Flow Direction

The data flow is from US Workstation to MUBM.

Data Type and Size

```
struct US_ControlMubm {
        uchar   CtrlfsyncMaxNoBitSlips;
        uchar   CtrlfsyncMaxNoBitErrors;
        uchar   CtrlfsyncPatternLength;
        uchar   CtrlfsyncUnlockThreshold;
        uchar   CtrlfsyncLockThreshold;
        uchar   Ctrldummy[27];
};
```

Data Rate and Frequency

The parameters are normally set once when starting the system. No data rate is to be defined.

3.3.5 Monitoring MUBM Stream

Description

The following defines the structure of data to monitor the baseband processing and different other MUBM internal parameters. The timer value included in this structure can be used to synchronise the MUBM time to the computer time. More detailed monitoring information is included in the VCDU data stream.

The monitoring information provided in this data stream will also be used for the MUBM front panel display, which is also available at the site of the antenna for antenna adjustment.

Data Flow Direction

The data flow is from MUBM to US Workstation.

Data Type and Size

```
struct US_MonitoringMubm {
        uint32 TimeStamp;

        /* monitoring data basebandProc */
        int32    monInpSigOffsFreq;
        int32    monBitrateOffsFreq;
        uint32   monViterbiDecBER;
        unit32   monInpSigAGC;
        uint32   monBaseSigAGC;
        uint32   freeSpaceBpBitstreamBuf;
        uint16   monBSFlags;
        char     monEbN0
        uchar    monBPdummy[357];
```

```
        /* monitoring data xritDec */
        uint32   monCorrBitErr;
        uint32   monCorrByteErr;
        uint32   monxStPfl;
        char     monBitSlip;
        uchar    monFSyncErr;
        unit16   monVCDUFlags;
        uint32   freeSpaceXdBitstreamBuf;
        uint32   freeSpaceXdVcduBuf;
        uchar    monXDdummy[360];

        /* monitoring data mubmCtr
        unit32   freeSpaceMcLogMessageBuf;
        unit32   allocSpaceMcLogMessageBuf;
        uchar    monMCdummy[244];

};
```

Data Rate and Frequency

The monitoring parameters are normally read regularly from the host. No data rate is to be defined. It shall be noted that the recalculation of the monitoring parameter is approximately the revolution rate of the MSG spacecraft, i.e. 0.6 seconds.

3.3.6 Log Data Stream

Description

The following defines the structure of one logging message of the MUBM. The message is coded using ASCII character. A detailed description is given in table

Data Name	Data Type	Data Description
XritSyncMarker[4]	Uchar	Synchronisation marker, refer to [MSI]
XritVCDU_data[892]	Uchar	VCDU data, refer to [MSI]
XritVCDU_RS[128]	Uchar	Reed Solomon check symbols, refer to [MSI]
XritVCDU_qual[208] {	Uchar	This data array comprises a packed structure of quality information, which is described in detail below.
T_bb_qual[8] {	Structure	Array of 8 times the following structure. The quality information is synchronised to the VCDU/CVCDU above. The synchronisation error is

Data Name	Data Type	Data Description
		max ±⅛ CVCDU for the Viterbi results. Please note that every element of this structure is doubled due to the fact that a Viterbi measurement is only possible every 4000 symbols. Due to this the symbol error rate is $\frac{1}{8 \cdot 4000} \sum_{8} ViterbiDecBER$
InpSigOffsFreq	Int32	NCO value of demodulator PLL offset frequency: $f_{c,offset}$ = InputSignalOffsetFrequency * 0.01 Hz
BitrateOffsFreq	Int32	NCO value of bit synchroniser PLL offset frequency: $f_{b,offset}$ = BitrateOffsetFrequency * 0.01 Hz
ViterbiDecBER	Uint16	Measured symbol error rate per 4000 symbols; due to the function of the Viterbi Decoder IC [Q1900]
BsFlags	uint16	Demodulator bitsynchroniser and Viterbi flags. Bit map of following flags (error/not locked if bit is set)
	bit0 bit1 bit2 bit3 bit4 bit5-bit16	- Demodulator lock - Bit synchroniser lock - Input level too low - Input level too high - Viterbi node sync lock - Reserved for future use
InpSigAGC	Unit32	Hex value of the input AGC on the I960-2nd DC, controlled by the basebandProc: $V_{Amplifier}$ = inpSigAGC * 0.32 dB InpSigAGC range: 127 …256 (delta = 40 dB) InpSigAGC max=255: Signal lost InpSigAGC max=120: Signal level overflow
BaseSigAGC	Unit32	BASEBAND SIGNAL AGC INSIDE THE HARRIS [HSP50214].
Dummy[4]	Uchar	Dummy bytes to fill up the structure to 4 byte boundary
}		
BitErr[4]	Uchar	4 bit error counters indicating the number of corrected bits (one counter per Reed-Solomon block) range: 0 … 128 set to 0 when RS fails

Data Name	Data Type	Data Description
ByteErr[4]	Uchar	4 byte error counters indicating the number of corrected bytes (one counter per Reed-Solomon block) range: 0 … 16 set to when RS fails
TimeStamp	Unit32	MUBM timestamp (2 ms modulo counter) at baseband processing stage
SyncErrs	Uchar	Bit errors in synchronisation marker 0x00 to maximum allowed bit errors. Sync pattern was found in datastream and value represents number of bit errors in sync pattern. A bigger value determines sync pattern not found.
BitSlip	Char	Sync misalignment to expected location invalid if no sync found (refer to VCDUFlags)
VCDUFlags	uint16 bit0 bit1 bit2 bit3 bit4 . . bit5 – bit13 bit14 bit15	Bit map of following flags (error occurred if bit is set) - RS error flag 0 - RS error flag 1 - RS error flag 2 - RS error flag 3 RS error flag is set with an uncorrectable error - frame sync flag (set when out of lock) Note: If frame sync not locked, RS decoding is not performed. Accordingly the corresponding quality information is not valid. Reserved for future use FS_NOTFOUND flag: Set if no sync was found but data is processed due to configured fly-wheel operation. FS_INV flag: Set if data is inverted due to an inverted sync pattern
}		
SkuCommandData[]	Uchar	This character array contains the SKU command data, specified in the SKU ICD [FICD/102]
SkuReturnData[]	Uchar	This character array contains the SKU return data, specified in the SKU ICD [FICD/102]

Data Name	Data Type	Data Description
CtrlFsyncMaxNoBitSlips	Uchar	Control parameter defining the allowed number of bit (±)slips of the frame synchroniser Range: 0 – 5 Default: 3
CtrlFsyncMaxNoBitErrors	Uchar	Control parameter defining the allowed number of bit errors in the frame synchronisation marker Range: 0 – 8 Default: 3
CtrlFsyncPatternLength	Uchar	Control parameter defining the length of the synchronisation marker: Default: 32
CtrlFsyncUnlockThreshold	Uchar	Control parameter defining the fly wheel parameter of the frame synchroniser (number of unfound fsync markers before frame sync lock loss is detected): Range: 1 – 20 Default: 3
CtrlFsyncLockThreshold	Uchar	Control parameter defining the 2nd fly wheel parameter of the frame synchroniser (number of found fsync markers before frame sync lock is detected): Range: 1 – 20 Default: 3
Ctrldummy[27]	Uchar	Undefined data for future use.
		Monitoring data The monitoring data mon* are calculated from the corresponding values attached to the xritVCDU data. These data are averaged, accumulated or traced over one satellite revolution, which yields to 0.6 sec. As the evaluation of monitoring parameter is linked to the dissemination of VCDUs, the following parameter are calculated over 11 VCDUs for LRIT and 85 VCDUs for HRIT.
MonTimeStamp	Uint32	MUBM 2 msec timer, modulo 2^{32} bit counter; actual timer value is actualised before the monitoring data are transferred via the SCSI bus
MonInpSigOffsFreq	int32	Average NCO value of demodulator PLL offset frequency: $f_{c,offset}$ = InpSigOffsFreq * 0.01 Hz
MonBitrateOffsFreq	int32	Average NCO value of bit synchroniser PLL offset

Data Name	Data Type	Data Description
		frequency: $f_{c,offset}$ = BitrateOffsFreq * 0.01 Hz
MonViterbiDecBER	Uint32	Average measured symbol error rate per 11 or 85 times 16000 symbols; due to the function of the Viterbi Decoder IC [Q1900]
MonInpSigAGC	Uint32	Average hex value of the input AGC on the I960-2nd DC, controlled by the basebandProc: $V_{Amplifier}$ = inpSigAGC * 0.32 dB InpSigAGC range: 120 …256 (delta = 40 dB) InpSigAGC max=255: Signal lost InpSigAGC max=120: Signal level overflow
MonBaseSigAGC	Uint32	AVERAGE BASEBAND SIGNAL AGC INSIDE THE HARRIS [HSP50214]
FreeSpaceBpBitstreamBuf;	Uint32	Defines the free space of the bitstream buffer on the baseband module in number of elements of the BitstreamBuffer on the baseband processing module. Each element contains 4 KB of bitstream data plus the corresponding quality information. Maximum value 208
MonBSFlags	Uint16	A detailed description of the single bits is given in BSFlags. Whenever a bit was set in the BSFlags during the monitoring time, the corresponding bit is set in monBSFlags
MonEbN0	Char	Eb/No calculated on base of Viterbi bit error rate; please note that this value is an estimation based on short term measurements.
MonBPdummy[357]	Uchar	Filler characters.
MonCorrBitErr	Uint32	Number of RS corrected bit errors
MonCorrByteErr	Uint32	Number of RS corrected byte errors
MonxStPfl	Uint32	Number of lost VCDUs due to frame sync loss or RS failure; cyclic counter which is not reset at the begin of a measurement
MonBitSlip	Char	Maximum sync misalignment to expected location during measurement period. Invalid if no sync found (refer to VCDUFlags)
MonFsyncErr	Uchar	Number of bit errors in frame sync marker over measurement period
MonVCDUFlags	Uint16	A detailed description of the single bits is given in VCDUFlags. Whenever a bit was set in the VCDUFlags during the monitoring time, the corresponding bit is set in monVCDUFlags
FreeSpaceXdBitstreamBuf	Uint32	Defines the free space of the bitstream buffer on

Data Name	Data Type	Data Description
		the Xrit decoder. The value is given in numbers of elements of the BitstreamBuffer. Each element contains 4 KB of bitstream data plus the corresponding quality information. Maximum value 104
FreeSpaceXdVcduBuf	Uint32	Defines the free space of the VCDU buffer on the Xrit decoder. The value is given in numbers of elements of the VCDU Buffer. Each element contains one XritVcdu structure. Maximum value 412
MonXDdummy[360]	Uchar	Filler characters
FreeSpaceMcLogMessageBuf	Uint32	Defines the free space of the log message buffer on the MUBM control module in number of log messages. Maximum value: 512
AllocSpaceMcLogMessageBuf	Uint32	Defines the number log messages in the log message buffer on the MUBM control module. Maximum value: 512
MonMCdummy[244]	Uchar	Filler characters
		Log messages of the MUBM are composed of the following character fields:
LogSeverity[2]	Char	'E ' error 'W ' warning 'I ' information 'T ' trace
ModuleId[4]	Char	'BPR ' basebandProc module 'XRD ' xritDec module 'MCB ' mubmCtr_basebandProcTask 'MCX ' mubmCtr_xritDecTask 'MCP ' mubmCtr_pnGenTask 'MCK ' mubmCtr_skuTask 'MCS ' mubmCtr_scsiTask
MubmTimer[9]	Char	"<ASCII coded 2ms timer value, 8 digits>' '
Message[49]	Char	"<ASCII message text, 48 characters followed by <LF>"

Table 1: Data Definition.

Data Flow Direction

The data flow is from MUBM to US Workstation.

Data Type and Size

```
struct US_MubmLogMessage {
        char logSeverity[2];
        char moduleId[4];
        char mubmTimer[9];
        char message[49];
};
```

Data Rate and Frequency

The log messages are stored in a cyclic buffer. Old log messages are overwritten if they are not retrieved from the host system. ~~The maximum data rate is TBD.~~No maximum data rate is defined.

4 Protocol

4.1 Overview

The MUBM system supports vendor unique commands dedicated to the main tasks, vendor unique commands dedicated to more general aspects of the MUBM monitor and control and mandatory commands specified in the [SCSI] standard.

The dedicated tasks are as followed:

- Retrieve VCDU reception data from the baseband processor and decoder
- Retrieve Pseudo Noise Key from the built in SKU
- Control SKU

General but vendor unique aspects are:

- Control MUBM
- Retrieve monitoring data
- Retrieve logging data

All other aspects of MUBM monitoring and control are covered by [SCSI] in form of mandatory commands.

4.2 Definition

The SCSI protocol is completely defined in [SCSI]. The command structure and the coding principles are according to this standard. For the coding of the commands dedicated to the SKU communication, please refer to DADF ICD SKU [FICD/102].

4.3 Method of Use

The following sections describe how to use the vendor unique commands to support the required functionality of the MUBM. It shall be noted that dedicated commands are implemented using different SCSI logical units (LUNs). This is done to allow different workstation processes initiate independent commands.

LUN 0: mandatory commands

LUN 1: reception of VCDUs

LUN 2: SKU commanding

4.3.1 Receive VCDUs

The command “Read Buffer – Monitoring MUBM” could be used to retrieve the status of the reception buffers, i.e. the number of processed VCDU data and the size of the corresponding buffer. Herewith the allocation length can be set-up for the command “Read Buffer – VCDU”, which then reads the received data in multiples of US_XritVcdu.

Due to the fact that the LRIT/HRIT data stream to the MUBM is of constant rate, the command “Read Buffer – VCDU” could also be used in another way. The command is able to support different reconnect times. Herewith it is possible to submit the command to the MUBM with an appropriate maximum reconnect time without knowing the available data size. Doing so, the MUBM will transfer either immediately available data up to the requested number to the host, or will wait for the maximums disconnect time.

If data becomes available in this time period, they are transmitted to the host, if not, the command is ended with a good status without returning data.

4.3.2 SKU Commands

Two commands are implemented for SKU communication:

- Write buffer SKU
- Read buffer SKU

The first one is used to write SKU command strings to the SKU via the MUBM implementing a serial interface to the SKU. Each SKU command returns data, either real data or status information. Disregarding the SKU return data, the user shall retrieve the SKU data using the read buffer SKU command, each time the write buffer SKU command was initiated before.

4.3.3 Monitor And Control MUBM

To monitor and control the MUBM system, three vendor unique commands are implemented dedicated to the special functions. The command

- Write buffer Control Baseband Process

is used to set-up different parameters of the baseband processing. Default parameter will be used when this command is not used.

To monitor the MUBM the command

- Read buffer monitoring MUBM

can be used. This command can be performed regularly to check the status of MUBM internal buffers and the state of the different processing activities.

The MUBM internal tasks write dedicated log messages to be retrieved by the command

- Read buffer MUBM log message.

4.4 Command Set

4.4.1 TEST UNIT READY (0x00)

This command checks whether the MUBM is ready.

CDB byte	7			Bit				0	Description
0	0	0	0	0	0	0	0	0	op code 0x00
1	L	U	N	0	0	0	0	0	lun 00 to 02 supported
2	0	0	0	0	0	0	0	0	Reserved 0
3	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0	Control byte

Table 2: CDB Test Unit Ready

Parameters There are no parameters for this command

Completion GOOD (00)

Status The command has been completed successfully and the preprocessor is ready for operation.
CHECK CONDITION (02)
The command failed. The REQUEST SENSE command shall be used to determine the reason for the failure.

Description This command is for compatibility only. The preprocessor is ready whenever it is accessible via the SCSI interface.

4.4.2 REQUEST SENSE (0x03)

This command retrieves sense data from MUBM.

CDB byte	7			bit				0	Description
0	0	0	0	0	0	0	1	1	op code 0x03
1	L	U	N	0	0	0	0	0	lun 00 to 02 supported
2	0	0	0	0	0	0	0	0	Reserved
3	0	0	0	0	0	0	0	0	
4	0	0	0	0	1	1	1	0	allocation length 14 (dec) byte
5	0	0	0	0	0	0	0	0	control byte

Table 3: CDB Request Sense

Data field byte	7			bit				0	Description
0	1	1	1	1	0	0	0	0	standard, current errors
1	0	0	0	0	0	0	0	0	SSSS: 4 bit sense key, see below
2	0	0	0	0	S	S	S	S	
3	m	S	b						information bytes set to zero
4									
5									
6						L	s	b	
7	0	0	0	0	0	1	1	0	6 bytes follow
8	m	S	b						command specific information set to zero
9									
A									
B						L	s	b	
C	a	A	a	a	a	A	a	a	8 bit additional sense code, see below
D	g	G	g	g	g	G	g	g	8 bit additional sense code qualifier, see below

Table 4: Return Data Request Sense

Parameter allocation length
specifies how many bytes are allocated to receive sense data. The MUBM will not send more than 14 bytes. Must not be less than 4 bytes.

Completion GOOD (00)

Status The command has been completed successfully
CHECK CONDITION (02)
The command failed. The sense information initially to be retrieved is lost. Another call to REQUEST SENSE can be used to obtain information about the failure.

Description The preprocessor sends min {allocation length, 14} bytes of sense data, having the above defined structure.

Refer to the list of sense information for the sense key (K), additional sense code (ASC) and the additional sense code qualifier (ASCQ) returned for the implemented commands.

Sense (hex)			Command (hex) (S=success, F=failure)							Description
K	ASC	ASCQ	00	03	0A	12	1D	3B	3C	
00	00	00	S	S	S	S	S	S	S	No specific condition
05	20	00								Invalid Command Operation Code
05	24	00	F	F	F	F	F	F	F	Invalid field in CDB
06	29	00	F	S	F	F	F	F	F	Reset occurred
B	47	00	F	F	F	F	F	F	F	SCSI parity error will be returned for other protocol errors as well
05	25	00	F	F	F	F	F	F	F	Unsupported logical unit
05	XX	00						F	F	See separate table below

Table 5: Sense Data Description

Sense (hex), K = 05, ASCQ = 0, ASC = see below	Command/Buffer ID (hex/hex) (S = success, F = Failure)						
	3B		3C				
	83	85	80	84	86	87	Buffer ID
C0							Invalid Buffer ID, if none of ID's above
04	F			F			Logical Unit not ready SKU is busy
C4	F						Write to SKU before reading last SKU result
C3				F			Timeout of SKU occured

Table 6: Vendor Unique Sense Data Description

4.4.3 SEND MESSAGE (0Ah)

CDB byte	7			bit				0	Description
0	0	0	0	0	1	0	1	0	op code 0 x 0A
1	L	U	N	0	0	0	0	0	lun 00 to 02 supported
2	msb								Allocation length
3									
4								lsb	
5	0	0	0	0	0	0	0	0	control byte

Table 7: CDB Send Message Command

Parameters Allocation Length: Specifies the allocation size in Bytes. Up to 32 KB may be transferred. Logical Unit Number: CDB Byte 1 / Bit 5 shall be set to 0 for LUN 1.

Completion GOOD (00)

Status The command has been completed successfully
CHECK CONDITION (02)
The command failed. The REQUEST SENSE command shall be issued to determine the reason for the failure.

Description The command is implemented for compliance to ANSI. Data sent is discarded For sending of data command 3Bh is used.

4.4.4 INQUIRY (0x12)

Retrieve INQUIRY data from the preprocessor.

CDB byte	7			bit				0	Description
0	0	0	0	1	0	0	1	0	op code 0 x 12
1	L	U	N	0	0	0	0	0	LUN 00 to 02 supported
2	0	0	0	0	0	0	0	0	Reserved
3	0	0	0	0	0	0	0	0	
4	0	0	1	1	1	0	0	0	Allocation length 56 (dec) bytes
5	0	0	0	0	0	0	0	0	control byte

Table 8: CDB Inquiry Command

Parameters allocation length
specifies how many bytes are allocated to receive inquiry data. The command will not return more than 56 bytes. Must not be less than 4.

Completion GOOD (00)

Status The command has been completed successfully.
CHECK CONDITION
The command failed. The REQUEST SENSE command shall be used to determine the reason for the failure.

Description The command returns 56 bytes of inquiry data, but not more than allocated.

The inquiry data has the following format:

Data field byte (dec)	7			bit				0	Description
0	0	0	0	0	C	0	P	1	The device type is “communication”, so C (bit3) is set, P (bit1) is not set
1	0	0	0	0	0	0	0	0	no removable medium
2	0	0	0	0	0	0	1	0	ANSI version is SCSI-2
3	0	0	0	0	0	0	0	0	no asynchronous event notification
4	0	0	1	1	0	0	1	1	51 (dec) data bytes follow
5	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	0	0	0	
7	0	0	0	1	0	0	0	0	8 bit bus, synchronous
8 • • • 15									Vendor identification 8 characters "VCS"
16 • • • 31									Product identification, 16 characters "MUBM HRIT” or “MUBM LRIT”
32 33 34 35									Product revision level, 4 characters e. g. "1.1"
36 37 38 39				High Nibble Low Nibble					SKU station number hexadecimal ASCII coded ('0'…'9', 'A'…'F') ~~[TBC]~~
40 41 42 • • • 55									filled to end with ASCII space (2x20)

Table 9: Inquiry Data Format

4.4.5 Send Diagnostic (0x1D)

Initiate self-test of the MUBM.

CDB byte	7			Bit				0	Description
0	0	0	0	1	1	1	0	1	op code 0 x 1D
1	L	U	N	0	0	1	0	0	lun 00 to 02 supported
2	0	0	0	0	0	0	0	0	Reserved
3	0	0	0	0	0	0	0	0	Allocation length set to zero
4	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0	control byte

Table 10: CDB Send Diagnostic Command

Parameters None **Completion** GOOD (00)

Status The command has been completed successfully.
CHECK CONDITION (02)
The command failed. The REQUEST SENSE command shall be used to determine the reason for the failure.

Description A self test is performed. When the test fails, the sense key HARDWARE ERROR can be retrieved by REQUEST SENSE.

4.4.6 WRITE BUFFER (0x3B)

Write to different buffers of the MUBM.

CDB byte	7			bit				0	Description
0	0	0	1	1	1	0	1	1	op code 0 x 3b
1	L	U	N	0	0	0	0	1	lun 0 to 2 supported, mode is vendor specific
2	Msb							lsb	buffer ID (see below)
3	Msb								
4									buffer offset (see below)
5								lsb	
6	Msb								
7									Allocation length (see below)
8								lsb	
9	0	0	0	0	0	0	0	0	control byte

Table 11: CDB Write Buffer

Parameters Buffer ID

Buffer offset and allocation length depends on buffer ID (see description below)

Completion GOOD (00)

Status The command has been completed successfully.
CHECK CONDITION (02)
The command failed. The REQUEST SENSE command shall be used to determine the reason for the failure.

Description Purpose of the command depends on buffer ID (see below)

4.4.6.1 Write Buffer-SKU (Buffer ID 0x83)

CDB Parameter

LUN: 0x2
Buffer ID: 0x83
Buffer Offset: 0
Allocation length: 10 to 100 ~~[TBC]~~

Transmit Data

Data field byte	Contents
0 to n	SKU_command_string

Table 12: Transmit Data Write Buffer SKU

Method of Use

This command is used to provide the SKU command string via the MUBM to the SKU. For detailed specification of the command strings, please refer to the DADF ICD SKU [FICD/102]. In principle each command sent to the SKU will result in a SKU response which shall be retrieved via the read buffer SKU command.

The normal sequence of commands is:

Write Buffer SKU

followed by:

Read Buffer SKU.

Please see read buffer SKU in section 4.4.7.2.

Performance Aspects

The average rate for this command cannot be specified. Generally, this command is used to initiate the PNK calculation. For this, the file rate on the HRIT or LRIT link is important.

Exception Handling

Check condition status are returned in the following cases:

- Invalid field in CDB
- Communication error to the SKU
- No SKU connected to the MUBM.

Busy condition is returned in the following case:

- The previous SKU command string is not yet forwarded to the SKU.

4.4.6.2 Write Buffer-Control Baseband Process (Buffer ID 0x85)

CDB Parameter

LUN: 0x0
Buffer ID: 0x85
Buffer Offset: 0
Allocation length: ~~TBD~~32

Transmit Data

Data field byte	Contents
~~0 to TBD~~32	US_ControlMubm

Table 13: Transmit Data Control Baseband Process Buffer

Method of Use

This command is used to set-up the parameter for the baseband control. The command is processed immediately. ~~The control data are distributed within the MUBM at regular time intervals. The interval is TBD msec.~~

Performance Aspects

No data rate is to be specified. Normally the parameters are set once at system start. Default parameters are set within the MUBM. Therefore this command is not necessary to start the MUBM baseband processing.

Exception Handling

Check condition status are returned in the following cases:

- Invalid field in CDB
- Invalid field in buffer

4.4.7 READ BUFFER (0x3C)

This command reads different buffers from the MUBM.

CDB byte	7			bit				0	Description
0	0	0	1	1	1	1	0	0	op code 0 x 3c
1	L	U	N	0	0	0	0	1	lun 0 to 2 supported, mode is vendor specific
2	Msb							lsb	buffer ID (see below)
3	Msb								buffer offset (see below)
4									
5								lsb	
6	Msb								Allocation length (see below)
7									
8								lsb	
9	0	0	0	0	0	0	0	0	control byte

Table 14: CDB Read Buffer Command

Parameters Buffer ID

Buffer offset and allocation length depends on buffer ID (see description below)

Completion GOOD (00)

Status The command has been completed successfully.
CHECK CONDITION (02)
The command failed. The REQUEST SENSE command shall be used to determine the reason for the failure.

Description Purpose of the command depends on buffer ID (see below)

4.4.7.1 Read Buffer - VCDU (Buffer ID 0x80)

CDB Parameter

LUN: 0x1

Buffer ID: 0x80

Buffer Offset: 0 to 3
This parameter defines the maximum disconnect duration
0: The MUBM processes the command immediately.
1: The MUBM allows up to 3 seconds to return data. ~~[TBC]~~
2: The MUBM allows up to 10 seconds to return data. ~~[TBC]~~
3: The MUBM allows up to 30 seconds to return data. ~~[TBC]~~

Allocation length: 1232 to 39424
This parameter defines the maximum number of bytes provided by the MUBM. The MUBM always returns integer multiples of US_XritVcdu.

Return Data:

Data field byte	Contents
0 to 1231	US_XritVcdu [n]
1232 to 2463	US_XritVcdu [n+1]
To	
(m-1) x 1232 to m x 1232 – 1	US_XritVcdu [n+m-1]

Table 15: Return Data Read Buffer VCDU

Method of Use

This command is used to retrieve LRIT or HRIT data from the MUBM. Due to the packet size of US_XritVcdu, it is recommended to use multiples of the packet size.Two methods for retrieving data are possible. One is to use the disconnect/reconnect features of the SCSI protocoll. The set-up of the maximum disconnect duration allows the host system to check for availability of data and retrieve data without polling the system. If no data are available, good status is returned. Whenever data are available, the MUBM will reconnect to the hsot and transfer up to the requested number of data.The disconnect duration parameter could also be used to wait the specified time for the collection of data. The size of the VCDU FIFO buffer is

38,5 KByte

In normal state, i. e. the MUBM receives VCDUs at a constant rate the buffer is filled up. To avoid loss of data caused by buffer overflow (the oldest VCDU is overwritten), the host has to read the data at an appropriate average data rate (refer to performance aspects).

The other method is to determine the appropriate allocation length and than to use this command to retrieve an available data set.

Performance Aspects

The same performance aspects apply as in section 3.3.1.

Exception Handling

Check condition status are returned in following cases:

- Invalid field in CDB
- VCDU buffer overflow

4.4.7.2 Read Buffer – SKU (Buffer ID 0x84)

CDB Parameter

LUN: 0x2

Buffer ID: 0x84

Buffer Offset: 0 to 3
This parameter defines the maximum disconnect duration
0: The MUBM processes the command immediately.
1: The MUBM allows up to 3 seconds to return data. ~~[TBC]~~
2: The MUBM allows up to 10 seconds to return data. ~~[TBC]~~
3: The MUBM allows up to 30 seconds to return data. ~~[TBC]~~

Allocation length: 10 to 100 ~~[TBC]~~
This parameter shall be set to the maximum size of the SKU return data specified in [FICD/102].

Return Data:

Data field byte	Contents
~~[TBD]~~refer to [FICD/102]	SKU_return_string

Table 16: Return Data Read Buffer SKU PNK

Method of Use

This command is used to retrieve the SKU return string via the MUBM from the SKU. For detailed specification of the return data strings, please refer to the DADF ICD SKU [FICD/102]. In principle each command sent to the SKU will result in a SKU response which shall be retrieved by this command.

The normal sequence of commands is:

Write Buffer SKU

followed by:

Read Buffer SKU.

Setting up the disconnect-duration allows the host system to write the SKU command via the write buffer SKU command and immediately to initiate the corresponding read buffer SKU command. The MUBM will either return the SKU return string immediately or will wait up to the specified maximum disconnect time. In the case no data are available, the command will be completed by a good status without returning data.

Please see write buffer SKU in section 4.4.6.1.

Performance Aspects

Please see performance aspects in section 4.4.6.1.

Exception Handling

Check condition status are returned in the following cases:

- Invalid field in CDB
- No SKU connected to the MUBM.

4.4.7.3 Read Buffer – Monitoring MUBM (Buffer ID 0x86)

CDB Parameter

LUN: 0x0

Buffer ID: 0x86

Buffer Offset: 0

Allocation length: ~~32 [TBC]~~1024
This parameter defines shall be fixed to the specified size.

Return Data:

Data field byte	Contents
0 to ~~TBD~~1024	US_MonitoringMubm

Table 17: Return Data Read Buffer Monitoring MUBM

Method of Use

This command is used to retrieve the monitoring data of the baseband process including status information of different MUBM internal buffers.

Performance Aspects

The US workstation software controls the data rate. The recommended value is ~~TBD~~ 1000 to 5000 msec.

Exception Handling

Check condition status are returned in the following cases:

- Invalid field in CDB

4.4.7.4 Read Buffer – MUBM Log Messages (Buffer ID 0x87)

CDB Parameter

LUN: 0x0

Buffer ID: 0x87

Buffer Offset: 0

Allocation length: 64 to 16384 ~~[TBC]~~

This parameter defines the maximum number of bytes provided by the MUBM. The MUBM always returns multiples of US_MubmLogMessage.

Return Data:

Data field byte	Contents
0 to 63	US_MubmLogMessage[n]
64to 127	US_MubmLogMessage[n+1]
To	
(m-1) x 64 to m x 64 - 1	US_MubmLogMessage [n+m-1]

Table 18: Return Data Read Buffer MUBM Log Messages

Method of Use

This command is used to retrieve the log data from the MUBM. The number of log messages in the log message buffer of the MUBM can be retrieved by the read buffer monitor command.

The log messages are internally stored in a cyclic buffer. In the case that no data are retrieved, old log messages are overwritten.

Performance Aspects

The same performance aspects apply as in section 3.3.6.

Exception Handling

Check condition status are returned in the following cases:

- Invalid field in CDB

5 Hardware Interface

5.1 Standards

The SCSI interface of the MUBM is compliant to the [SCSI] standard. No further specification is therefore needed here.

5.2 Signals

The signals of the interface are specified in [SCSI]. The single-ended SCSI interface signals are implemented.

5.3 Plugs, sockets

A 50 pin, low-density device connector is used for the MUBM. Pin allocation and physical characteristics are according to [SCSI].

5.4 Electrical Specification

The single-ended version of SCSI is implemented. All electrical specifications are according to [SCSI].

6 Glossary

Refer to [FGLO]