BAE SYSTEMS

Advanced Technology Centre,
Systems Department

West Hanningfield Road,
Great Baddow,
Chelmsford,
Essex, CM2 8HN.

BSATC Ref: GSY/050042/107853 Issue 5 18 June, 2007

ESA Contract No: 18081/04/D/SW

IFMS

ICD for the IFMS ESU datasets

ESU_ICD_DSet

European Space Agency
CONTRACT REPORT
The work described in this report was performed under ESA contract. Responsibility for the content resides in the author or organization that prepared it.

Abstract:

Interface document for the IFMS External Storage Unit

Keywords

Delta-DOR, Enhanced open loop processor.

Changes since the issue 4

Date	Author	Change
2007-06-18	NDJ	Section 2.2 - Changed parameters in the ASCII header file to reflect the change in polarization naming from LHC/RHC to Y/X.

Approvals

The contents of this document have been formally approved by the following personnel.

Author
Name: Nick James

Proposal Manager	Quality Engineer	Issuing Authority
Name: Nick James	*Name:* S. Phillipo	*Name:* A.C. Baslington

Distribution

Ricard Abelló	ESA
IFMS project office	BSATC (originals)
Nick James	BSATC

Contents

1. INTRODUCTION AND BACKGROUND

This document describes the format of the datasets stored on the ESU. These datasets use data produced by the enhanced open loop IFMS and they are intended for use by a remote correlator which forms part of the delta-DOR system.

The binary ESU datasets are formatted by the IFMS and transported to the ESU via the IFMS Ethernet network ports LAN1 and LAN2 (J52 and J53) using UDP/IP datagrams.

1.1. References

1 Software User Manual, IFMS-SUM 11.0.0

2. ESU DATASETS

2.1. EOLP filenames

EOLP data-sets consist of an ASCII file containing the IFMS current configuration relevant to EOLP and multiple binary files containing the data.

2.1.1. Data-set file location.

All data-sets will be stored in a unique directory as specified below:

<data_set_root>/<dataset_file>

2.1.2. Data-set file naming.

The filename for files collected on the ESU contains fields separated by underscore "_" characters as shown in the following example:

NNO1_MEX3_2005_108_OP_E1_145513_0001

As indicated in the example, the nominal length of a filename is 31 characters The fields to be included in the filename are:

- Station ID 4 characters (*)
- Spacecraft or quasar ID 4 characters (†)
- Year 4 characters
- Day Of Year 3 characters
- Data-set Kind 2 characters (*)
- Data Type 2 characters (fixed value: "E1" or "E2", as "Extended OLP", GDSP 1 or 2)
- Acquisition Start Time 6 characters (format: "hhmmss")
- Data-set Sequence ID 4 characters

(*) The value of these fields is determined from the ESU configuration at the time of data-set creation.

(†) The value of this field is sent with the start command.

If the length of the string value of the corresponding configuration parameter is less than the length indicated above the fields are padded on the right with underscore "_" characters.

2.1.3. Data-set file format.

With the exception of the first file of the dataset EOLP data-sets are binary files as described in section 2.3. The first file of the group (sequence ID = 0000) will contain an ASCII extract of the IFMS active configuration at the time of data collection start.

The first data-set will contain in ASCII the source IFMS configuration, and the ESU configuration. The following data-sets will contain each an integer number of records as provided by the EOLP GDSP (1468 bytes each), and a new data-set will be created every e.g. 1 minute (i.e. 270 MB at maximum rate).

2.2. EOLP ASCII data format

The EOLP ASCII data format (file sequence ID = 0000) contains the relevant IFMS configuration at the start of the EOLP capture. This configuration is taken from the "modulator", "freqplan" and "enopenloop" configuration parts of the IFMS.

Data is stored in this file as ASCII records with a tag associated with each parameter. The fields are shown in Table 1. The active table is the active configuration of the IFMS. An example file is shown in Table 2.

```
  <station_id>              (from ESU configuration)
  <spacecraft_id>           (from ESU configuration)
  <dset_kind>               (from ESU configuration)
  <dap_type>                (will be "E1" or "E2")
  <internal_reference>      (from IFMS status)
  <uplink_carrier_230>      (from IFMS configuration)
  <actual_carrier_indic>    (from IFMS status)
  <actual_splrate_indic>    (from IFMS status)
  <active_table>            (extract of the current IFMS configuration)
```

Table 1. Fields in a sequence 0000 EOLP dataset file

```
<header>
<station_id>              BADW                                 </station_id>
<spacecraft_id>           tt08                              </spacecraft_id>
<dset_kind>               TS                                    </dset_kind>
<dap_type>                E1                                     </dap_type>
<internal_reference>      No                           </internal_reference>
<uplink_carrier_230>      No                           </uplink_carrier_230>
<actual_carrier_indic>    3049426780.                </actual_carrier_indic>
<actual_splrate_indic>    176.                       </actual_splrate_indic>
<active_table>
   UlmMode                       = "Normal"              ; // -
   UlmCarNomLvl                  = 0                     ; // dBm
   UlmCarTstLvl                  = 60.0                  ; // dB
   UlmCarSpecInv                 = No                    ; // -
   FreqUlmCarFrSel               = "70MHz Oper"          ; // -
   FreqUlmCarFrOffs              = 300000                ; // Hz
   FreqSpecInv                   = No                    ; // -
   FreqUplkConv                  = 1000000000            ; // Hz
   FreqCoherTrs                  = Yes                   ; // -
   FreqTR1                       = 1                     ; // -
   FreqTR2                       = 1                     ; // -
   FreqDnlkCF                    = 1070300000            ; // Hz
   FreqDnlkConv                  = 1000000000            ; // Hz
   EolpSampleRate                = 100000                ; // Hz
   EolpQuantisation              = "16bit"               ; // -
   EolpFixedGain                 = Yes                   ; // -
   EolpGainValue                 = 90                    ; // dB
   EolpYSrcOffset                = 0                     ; // Hz
   EolpXSrcOffset                = 0                     ; // Hz
   EolpAuxSrcOffset              = 0                     ; // Hz
   EolpSubCCentreFreqOffset      = 0                     ; // Hz
   Eolp1SubC0Source              = "X"                   ; // -
   Eolp1SubC1Source              = "Y"                   ; // -
   Eolp1SubC2Source              = "Y"                   ; // -
   Eolp1SubC3Source              = "Y"                   ; // -
   Eolp2SubC0Source              = "Y"                   ; // -
   Eolp2SubC1Source              = "Y"                   ; // -
   Eolp2SubC2Source              = "Y"                   ; // -
   Eolp2SubC3Source              = "Y"                   ; // -
   Eolp1SubC0FreqOffs            = 100000                ; // Hz
   Eolp1SubC1FreqOffs            = 0                     ; // Hz
   Eolp1SubC2FreqOffs            = 0                     ; // Hz
   Eolp1SubC3FreqOffs            = 0                     ; // Hz
   Eolp2SubC0FreqOffs            = 0                     ; // Hz
   Eolp2SubC1FreqOffs            = 0                     ; // Hz
   Eolp2SubC2FreqOffs            = 0                     ; // Hz
   Eolp2SubC3FreqOffs            = 0                     ; // Hz
</active_table>
</header>
```

Table 2. Example of a sequence 0000 EOLP dataset file

A description of the parameters in the ASCII header can be found in the IFMS-SUM [1].

2.3. EOLP binary data format

The EOLP binary data format (file sequence ID >= 0001) is identical to the UDP data passed over the LAN. Formatting is performed in the GDSP and each record is sent over the LAN encapsulated in a UDP/IP datagram.

The stored binary file consists of an arbitrary number of records. The record size is limited by the need to transport it within a UDP/IP datagram. The maximum Ethernet frame size is 1526 bytes consisting of a 22 byte Ethernet header, a 4 byte Ethernet checksum and 1500 bytes of payload. The IP header and UDP header use a further 28 bytes and a further 2 bytes of padding are included to ensure word alignment. This means that the maximum record size is 1470 bytes. This is made up of a header section and a data section. In the EOLP binary format the header is 19 words (76 bytes) long and this is followed by 87 data blocks of 16 bytes each. This gives an overall record length of 1468 bytes with 1392 data bytes per record.

The overall record format is described in section 2.3.3. The format of the data blocks is described in sections 2.3.1 and 2.3.2. The proposed file format allows for either multiplexed or single subchannel storage however the raw files from the GDSP will always be multiplexed. Single subchannel files will be generated in post processing if required.

In the following descriptions header words are numbered H00 to Hxx. Data blocks are numbered aa through ax then ba through bx etc.

2.3.1. Multiplexed subchannel data blocks

Binary datafiles recorded on the ESU are always in the multiplexed format. All four subchannels are stored multiplexed together in each block. The blocks consist of 32 nibbles numbered from 0 to 31 with the most significant nibble in each byte coming first. Each bit in the nibble encodes one of the subchannels. Within each block the subchannel bits are stored serially starting with the MSB of the real sample and ending with the LSB of the imaginary sample. The number of samples which can be packed into each 16-byte block is dependent on the requested quantization as shown in Table 3. Examples of the packing structure are shown in Figure 1. This structure arises from hardware implementation aspects in the GDSPs since the multiplexed data is stored exactly as it is received from the GDSPs.

Quantization	Channels	Bits/sample	Samples/block	Samples/frame
1	4	2×4 = 8	16	1392
2	4	4×4 = 16	8	696
4	4	8×4 = 32	4	348
8	4	16×4 = 64	2	174
16	4	32×4 = 128	1	87

Table 3. Data packing in a DDOR application frame

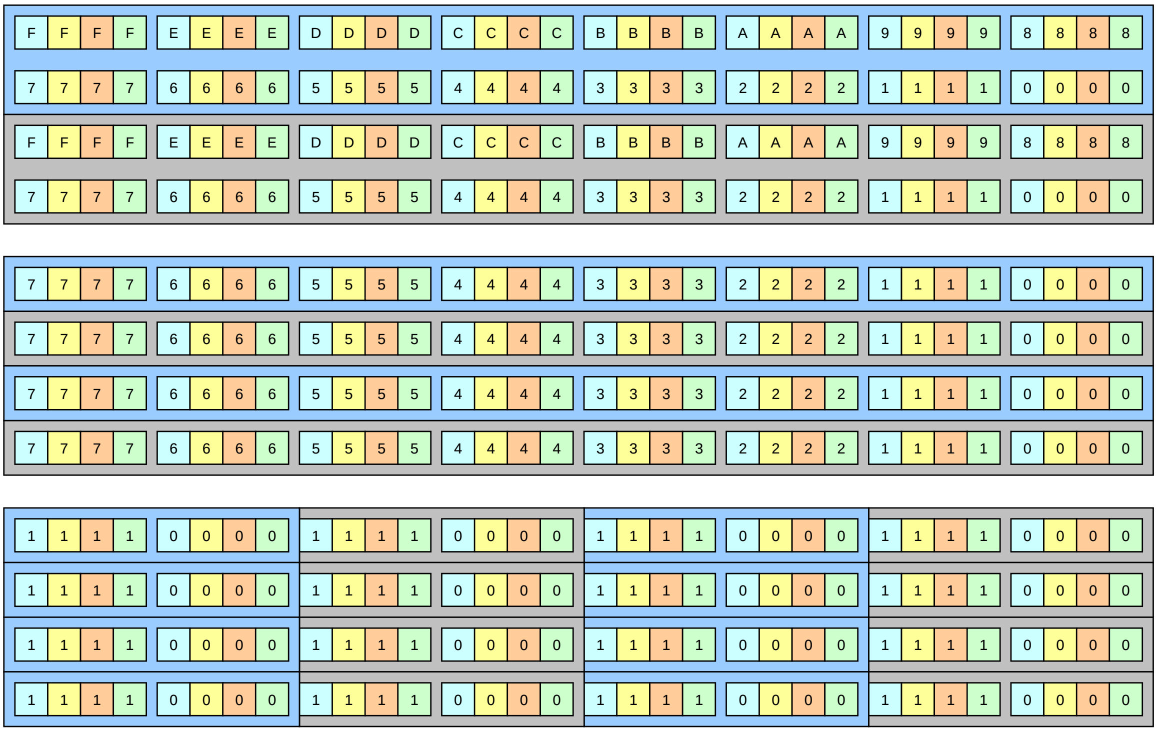

Figure 1. Example subchannel datablock packing for 16-bit (top), 8-bit (middle) and 2-bit (bottom) quantization. The four subchannels 0..3 are represented by the blue, yellow, orange and green blocks respectively. Bits in words are numbered in hexadecimal with the higher numbers being the more

significant bits. Real samples have a blue background and imaginary samples have a grey background. Each row is a 32-bit unsigned word with the MSB on the left and LSB on the right. Words are ordered from top to bottom. These blocks contain 1, 2 and 8 samples respectively.

2.3.2. Non-multiplexed channel data blocks

Note that this file format is not generated by the GDSPs but can be generated by post processing. Samples are packed into data blocks in the same order as for multiplexed blocks but four times as many samples can be stored. An example for 2-bit quantization and subchannel 1 is shown in Figure 2.

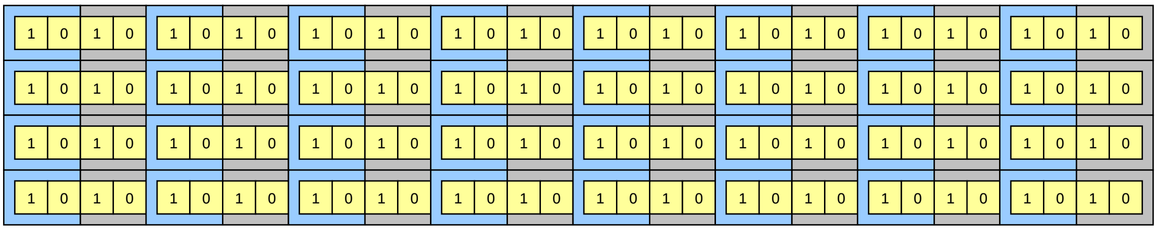

Figure 2. Example subchannel datablock packing for non multiplexed 2-bit quantization. Bits in words are numbered in hexadecimal with the higher numbers being the more significant bits. Real samples have a blue background and imaginary samples have a grey background. Each row is a 32-bit unsigned word with the MSB on the left and LSB on the right. Words are ordered from top to bottom. This block contains 32 samples.

2.3.3. Record format

ID	31							0
H00	0xA3C725B6							
H01	recordlength[31..16]	hdrlen[15..8]	blocksize[7..0]					
H02	samplerate[31..16]	cfegain[15..6]	qu[5:3]	msg[2:0]				
H03	Frameid[31..0]							
H04	Version[31..25]	timetag_samps[24..0]						
H05	Offsetfreq[31..0]							
H06	timetag_secs[31:15]	subc[14..11]	digitalgain[10..0]					
H07	subchan1_offset[31..0]							
H08	subchan2_offset[31..0]							
H09	subchan3_offset[31..0]							
H10	subchan4_offset[31..0]							
H11	sweeprate[31..0]							
H12	Path_delay[31..0]							
H13		hs[23]	scmr[22..11]	sweepchange[10..0]				
H14	ncov[31]	ncoreset_c[30..20]	ncoreset_t[19..0]					
H15								
H16								
H17								
H18								
aa_0	N00	N01	N02	N03	N04	N05	N06	N07
aa_1	N08	N09	N10	N11	N12	N13	N14	N15
aa_2	N16	N17	N18	N19	N20	N21	N22	N23
aa_3	N24	N25	N26	N27	N28	N29	N30	N31
ab_0	N00	N01	N02	N03	N04	N05	N06	N07
ab_1	N08	N09	N10	N11	N12	N13	N14	N15
ab_2	N16	N17	N18	N19	N20	N21	N22	N23
ab_3	N24	N25	N26	N27	N28	N29	N30	N31
dh_0	N00	N01	N02	N03	N04	N05	N06	N07
dh_1	N08	N09	N10	N11	N12	N13	N14	N15
dh_2	N16	N17	N18	N19	N20	N21	N22	N23
dh_3	N24	N25	N26	N27	N28	N29	N30	N31
di_0	N00	N01	N02	N03	N04	N05	N06	N07
di_1	N08	N09	N10	N11	N12	N13	N14	N15
di_2	N16	N17	N18	N19	N20	N21	N22	N23
di_3	N24	N25	N26	N27	N28	N29	N30	N31

Spare fields are grey. All spare fields should be ignored.

2.3.4. Record description

Identifier	Size	Range	Description
blocksize	8	0..255	Block size in bytes. Note that data blocks are described in section 2.3.1.
cfegain	10	0..650	CFE wideband gain in 0.1 dB steps.
digitalgain	11	0..2047	The digital gain of the EOLP path from GDSP input to LAN. Fixed point, 0.1 dB steps. TBD offset. This does not include the CFE gain.
frameid	32	0..4294967295	Frame counter. Increments by one for every transmitted frame. Wraps at 2^{32}.
gdspid	8	1..255	Serial number of source GDSP.
hdrlen	8	0..255	Length of the header in bytes
hs	1	0..1	Flag to indicate CFE mode (0=>narrowband, 1=>wideband).
magic	32	0xA3C725B6	This is intended as a unique identifier of the data record in both big and little endian environments so that programs can automatically reorder bytes as necessary
msg	3	6	Identifies this as an EOLP data record.
ncoreset_t	20	0 .. 863999	Subchannel NCO reset time in 0.1 s units relative to the start of the UTC day. Only valid if version $\geq$ 2. The true NCO reset time in seconds since the start of the day is given by: $ncoreset\_t/10.0 + ncoreset\_c/(70.0\times10^6)$
ncoreset_c	11	-1024 .. +1023	11-bit two's complement NCO reset time correction in units of 70 MHz clock periods. This correction is necessary to translate the time given by ncoreset_t to the actual reset time in the signal path. Only valid if the file version $\geq$ 2.
ncov	1	0..1	Set to 1 if the NCO reset time is valid. Set to 0 otherwise.
offsetfreq	32	-2147483647.. 2147483647	This is the offset applied to each subchannel centre frequency resulting from the Doppler predict (PDOP), frequency plan (FREQ) and the frequency offset (SUBO) commands. The offset applies to the first sample of the frame. This value is in machine units. To convert to conventional units multiply by $35\times10^6/2^{32}$ ($\approx$8.149 mHz).
path_delay	32	$0..2^{32}$-1	Path delay from GDSP connector interface to timetagging point in units of 35 MHz cycles.
qu	3	0..7	Sample quantization (0=>1 bit, 1=>2 bits, 2=>4 bits, 4=>8bits, 5=>16bits, 3, 6, 7 spare)
recordlength	16	0..65535	Record length in bytes.
samplerate	16	0..65535	The actual sample rate given by 17.5×10^6 Hz divided by this value. Note that the sample rate is constrained to be $17.5\times10^6/(8n)$ where n is an integer.

scmr	12		This word indicates the internal mapping of physical subchannel processors onto logical subchannels. There are five physical SCPs numbered from 0 to 4 (each represented in a 3-bit word). This 12-bit word has four 3-bit fields, one for each of the four logical subchannels. The physical SCP associated with SC0 is stored in the lower 3-bits and the physical SCP associated with SC3 is stored in the upper 3-bits. This data is for engineering/debug applications only.
subc	4	0..16	Subchannels in this file. 0=>all 4 subchannels multiplexed. 1, 2, 3, 4 => individual subchannel not multiplexed. >4 not valid.
subchan*_offset	32	-2147483647.. 2147483647	Subchannel centre frequency offset (from SUBC). Same format as offsetfreq.
sweepchange	11	0 .. 2047	Sample number in frame for sweep change. If this is zero there is no sweep change during the frame.
sweeprate	32	-2147483647.. 2147483647	This is the Doppler sweep rate applied during the frame (if sweepchange = 0) or from sample n (if sweepchange = n). The rate is in machine units. To convert to conventional units multiply by $(35\times10^6)^2/2^{58} \approx 4.25$ mHz/s.
timetag_samps	25	0..17499999	Timetag for the first sample in this frame in clock ticks since the last second marker. The clock tick frequency is 17.5×10^6 Hz. The timetag is applied at the LAN formatter. The delay from the input of the GDSP to this formatter is given by path_delay.
timetag_secs	17	0..86399	Timetag for the first sample in this frame in unit seconds since UTC midnight. This timetag is coincident with timetag_samps. The complete timetag is made up of the sum of timetag_secs and timetag_samps.
version	7	0..127	File version number. This will change at any time the file format or interpretation change.

2.3.5. Binary data mapping

The samples encoded in the data blocks are stored as binary words of 1, 2, 4, 8 or 16 bits quantization. The data is obtained by processing the 16-bit, two's complement samples produced by the subchannel FIR filter as follows:

1. The output of the subchannel filter is a 16-bit two’s complement word with equispaced quantization levels. This represents numbers in the range -32768 to +32767 and any sample outside this range overflows and wraps around the word.
2. The 16-bit samples are multiplied by 2 (i.e. shifted left by one place) to form a 17-bit word with 0 in the LSB. The resulting 17-bit samples are hard limited to be in the range -32768, +32767 and then the MSB is dropped so that the samples return to a 16-bit representation.
3. The word is right shifted by 16-q places (where q is the requested quantization) and the least significant q bits of the result are stored in the output file.

The mapping of the binary data to signal value is shown in Table 4. This mapping is chosen to keep the gain value of the system roughly constant for different values of quantization. The addition of half an LSB is needed in order to remove the DC offset effect of the truncation performed in step 3 above.

Note that in the case of 16-bit quantization the actual resolution is only 15 bits due to the left shift performed in step 2 above. This means that the LSB for 16-bit words will always be zero except for the case of positive saturation.

Quantization (n)	Word range (m)	Signal value
16	-32768 .. +32767	m+0.5
8	-128 .. +127	256m+128
4	-8 .. +7	4096m+2048
2	-2 .. 1	16384m+8192
1	-1 .. 0	32768m+16384
n = 1, 2, 4, 8, 16	$m = -2^{n-1} .. +(2^{n-1}-1)$	$2^{16-n} \times (m+0.5)$

Table 4. Data mapping

2.4. NOTES

2.4.1. Use of the magic identifier

The magic is a known 32-bit number (i.e. 4 bytes). It is included to allow records to be read in both little-endian and big-endian environments. The byte order in the records is big-endian (i.e. 32-bit unsigned ints are stored with the most significant byte first). If the header is read as a struct of unsigned ints on a little-endian machine such as Linux on Intel the magic would be reversed to 0xB625C7A3. This can be used to flag a swap of all the header bytes so that they read correctly.

2.4.2. Determination of subchannel RF centre frequency

The incoming RF carrier is downconverted to IF as follows:

IF_freq = RF_freq + Eolp<XXX>SrcOffset - FreqDnlkConv

where Eolp<XXX>SrcOffset and FreqDnlkConv are exact frequencies obtained from the ASCII header file (section 2.2). EolpXXXSrcOffset is a frequency offset which is unique to the subchannel source (i.e. Y, X or AUX). Both of these values will be constant for each dataset.

The centre frequency of the subchannel at the IF is given by:

IF_freq = 70MHz + offsetfreq + subchan<n>_offset

In this case offsetfreq and subchan<n>_offset are held in the binary file (section 2.3) as exact digital representations (Note that frequencies are not on a 1 Hz grid) relative to the 70 MHz IF centre frequency. The subchan<n>_offset variable is the specific offset for each subchannel and it will be constant for the duration of the dataset. The offsetfreq variable is the common subchannel frequency offset. This includes any Doppler presteering and so may be variable over a dataset.

The RF frequency of the subchannel centre is therefore:

RF_freq = 70MHz + offsetfreq + subchan<n>_offset + FreqDnlkConv - Eolp<XXX>SrcOffset

The correct <XXX> to use for each subchannel is determined by examining the source information recorded as Eolp<n>SubC<m>Source in the ASCII header file.

The sweeprate and sweepchange variables in the binary files are only relevant when real-time Doppler predicts are active. When computed in conventional units sweeprate gives the offset frequency to add to each sample in the frame relative to the start of frame frequency offset (offsetfreq). If the sweepchange flag is nonzero the sweep starts that many samples into the frame with the previous frame's sweep rate used up to that point. If sweepchange is zero the sweeprate is valid for the entire frame.

2.4.3. Correction of the sample timetag

The actual sample time is given by:

UTC = timetag_secs + timetag_samps*(1/17500000) - path_delay/35000000

These parameters are obtained from the binary dataset files. The raw timetag refers to samples at the output of the (variable delay) EOLP filter. The path delay corrects the timetag to a fixed point at the input to the EOLP processing.

2.4.4. Synchronization across multiple EOLPs

One EOLP can implement four subchannels. If more subchannels are required a second EOLP can be activated leading to two parallel EOLP datasets (E1 and E2). Sampling times are synchronized between the two datasets however the exact start time and, consequently, the time of the record boundaries is not synchronized.