DOCUMENT NUMBER: **TN-IMGSER**

DOCUMENT REVISION: **A1**

DATE: **25 March 2025**

CAGE CODE: **A0810**

GRIFO-E/600

GRIFO-E IMAGES SERIALIZATION OVER UDP TECHNICAL NOTE

REVISION RECORD

Rev.	Mod. N.	Date	Description	Author/s
A0	N/A	29 May 2024	DRAFT	
A1	**N/A**	**25 March 2025**	**Draft 1:** • **§5.5, Table 5-1: add missed fields PROT_VER and FRAG_SPARE** • **Fix various clerical error** • **Add §6.3 Note about image postprocessing for visualization**	

TABLE OF CONTETS

INDEX OF FIGURES

Non è stata trovata alcuna voce dell'indice delle figure.

INDEX OF TABLES

1 SCOPE

1.1 Identification

This Technical Note describes the protocol used to by the Radar GRIFO-E/600 to serialize images over Ethernet.

1.2 System overview

The GRIFO-E/600 Radar is a Fire Control Radar (herein after called "the Radar") that generate two type of images:

- High rate, low-resolution Radar operative display, herein after called MFD. On the MFD are displayed:
    - Synthetic symbols, as A/C navigation information, antenna scan information, acquisition cursor
    - Synthetic targets symbols
    - Raw beam map, DBS map and low-resolution SAR images background
- Low rate, high resolution Spot SAR images

The Radar shall serialize, at real-time, the produced images for post-processing by an image receiver equipment directly connected to the Radar via a standard Gigabit Ethernet local network.

1.3 Document overview

This purpose of this technical note is:

- §4: Define the image format
- §5: Define the serialization protocol

For the understanding of this document, it is required a well knowledge of the Radar functionalities, Radar operative display characteristics and Spot SAR image characteristics.

For a complete list of used acronyms see §6.1.

2 REFERENCED DOCUMENT

2.1 Applicable document

Ref.	Code	Title
[DECD]	ICD7069914 Rev. draft A	GRIFO-E/600 RADAR DATA EXCHANGE CONTROL DOCUMENT

Table 2-1: Applicable documents

2.2 Reference document

Ref.	Code	Title

Table 2-2: Referenced documents

3 FOREWORDS

The Radar image serialization format and Ethernet transport protocol described in this technical note, is based on the requirements and characteristics of the images produced by the Radar GRIFO-E/600.

Nevertheless, both image format and protocol include provision to potentially support more generic data serialization and more sophisticate transmission flow control and errors recovery.

The Radar serialization interface requirements are subdivided into:

- The Image serialization format (the "file" format), independent on the serialization protocol
- The serialization protocol, potentially independent on the image serialization format and applicable on other kind of file

The image serialization format:

- Include attributes that are constant for a given GRIFO-E/600 version (for instance, the image MFD geometry that is fixed at 484x484 pixels)
- Is subdivided into data table, that include special fields used to "tag" the data table contents and size (again, even if they are constant for a given GRIFO-E/600 version). The tags permit to maintain compatibility and:
    - Extend the image attribute, adding new, properly tagged, data block
    - Expand existing data block, increasing the size and changing the data block version
- Geographic localization data, even if not always fully applicable
- Palette colour map, event if not always applicable

The serialization protocol:

- Includes fields to support multiple flows of data over the same connection, intended as association of a transmitter (IPv4 address and UDP port) and a receiver (IPv4 address and UDP port).
- Includes a minimalistic flow control, controlled by the receiver
- Includes provision for implementing advanced flow control with potentially smart retransmission of lost packets

4 RADAR IMAGE FORMAT

4.1 Data field encoding format

In the next paragraphs, the data fields are encoded as follows:

Type	Size (bytes)	Encoding
U8	1	8 bits Unsigned integer
I8	1	8 bits signed integer, 2'n complement
E8	1	Encoded 8 bits value
U16	2	16 bits Unsigned integer
I16	2	16 bits Signed integer, 2'n complement
I32	4	32 bits Signed integer, 2'n complement
U32	4	32 bits Unsigned integer
F32	4	32 bits IEEE-754 single precision floating point
AF32	4	Angle, radiant, 32 bits IEEE-754 single precision floating point
TT32	4	Time tag, represented as a 32 bits unsigned integer with LSB=64 microseconds
GC32	4	Geographic coordinate, IEEE-754 single precision floating point, radiant: • Latitude: positive North • Longitude: positive East
structure	variable	A sequence of fields and sub-structure

For all the multi bytes encoding formats, the byte ordering (endianness) is **LITTLE ENDIAN.**

The notation:

- <T>[<N>] represent an array of <N> elements of type <T>.
- <T>[<N> <M>] represent an array on <N> array of <M> elements of type <T> (a matrix)
- Array indexes start from 0 to <N>-1
- The array and matrix serialization order start from the rightmost elements towards left. For instance, a matrix of [3][3] shall be serialized in the following order:
    - [0,0], [0,1], [0,2], [1,0], [1,1], [1,2], [2,0], [2,1], [2,2]

The two dimensions image matrix has the format [Y, X], where:

- X coordinate is the horizontal axis, the image columns.
- Y coordinate it the vertical axis, the image rows.
- The image picture element (pixel) first value is at [0,0] correspond to the left-up corner.
- The image second value correspond to pixel at [0,1].
- The image $(X+1)^{th}$ value correspond to the pixel at [1,0], second row, first columns, and so on.

- Where not specified, the XY plane is oriented to the image true heading.

The fields can be marker as:

- **Radar reserved** can assume every value and shall be ignored by the receiver.
- reserved: reserved for protocol or format enhancements, shall be fixed at zero.
- Spare: spare for the application level, cab be used n further expansion without broking the protocol. Shall be fixed at zero.

4.2 Image Basic Format

A Radar image is composed of:

- The **Image Leader**: a set of attributes, describing the image geometry and supplying other information required for proper image post-processing and visualization.
- The **Image body**: the image picture elements (pixels) values, composed by a string of bytes to be interpreted as a square matrix as defined in the image attribute.

The leader and body are serializable as a sequence of bytes, subdivided into data block, where each block serialize a specific set of information, prefixed by a block tag (TAG) carrying a raw description of the block.

The Table 4-1 show the sequence of data blocks, while Table 4-2 show the format of the block tags.

Offset (bytes)	Field	Size (bytes)	Description
LEADER			
0	HEADER_TAG	8	A TAG structure identifying the HEADER_DATA
8	HEADER_DATA	48	Contains the essential graphic attribute, as image geometry
56	GEO_TAG	8	A TAG structure identifying the following GEO_DATA
64	GEO_DATA	76	The Geographic localization data (WGS-84)
140	RESERVED_TAG	8	A TAG structure identifying the RESERVED_DATA
148	RESERVED_DATA	128	Radar Reserved data, shall be ignored by the consumer
276	CM_TAG	8	A TAG structure identifying the following COLOUR_MAP
284	COLOUR_MAP	1024	The image palette colour map, always present, also when no applicable (for instance, in case of grey scale image)
1308	PIXEL_TAG	8	A TAG structure identifying the following PIXEL_DATA
BODY			
1316	PIXEL_DATA	DY*STRIDE*BPP	The pixel samples, variable size in accordance with the attribute value of the field in HEADER_DATA: • DY: number of rows • STRIDE: number of serialized samples per row. Normally equal to number of valid columns (samples) per row • BPP: bytes per sample (1 for 8bits samples images, 2 for 16 bits)

Table 4-1: Image data blocks

4.2.1 Block TAG structure and relative fields offsets

The TAG structures (HEADER_TAG, GEO_TAG, RESERVED_TAG, CM_TAG, PIXEL_TAG) format is:

Offset (bytes)	Field	Type	Description
0	ID	U8[2]	A marker univocally identifying the data block
2	VALID	U8	Validity flag of the data block
3	VERSION	U8	Format version of the data block
4	SIZE	U32	Size of the data structure, expressed in bytes

Table 4-2: Block TAG structure format

In all the structure description tables, the field offset value is relative to the beginning of the table in the image record. For instance, TAG field offsets are relative to the beginning of the tag structure (the SIZE field of tag structure PIXEL_TAG is at position 1316+4 from the beginning of the image record - Table 4-1).

The TAG structures are a provision for format expansion (add info to existing blocks) and extension (add new data blocks).

In case of changes in the block size, the relevant block tag SIZE field changes and as well as change the offset of all the subsequent data block. The beginning of a block tag and attached data block can be run-time calculated as;

- Block TAG: sum of the SIZE of all the preceding data TAG.
- Block data: Block TAG + 8 bytes.

For instance, if HEADER_DATA grow of 4 bytes:

- HEADER_TAG.SIZE grow of 4, that is change from 48 from 56.
- GEO_TAG start offset changes from 64 to 72.
- In accordance, all subsequent offsets are incremented by 4.

4.3 Radar image data contents

The detailed image record layout is described in the following table. The column offset contains the offset from the first byte of the record, calculate in accordance with actual blocks size (see §4.2.1). Where offset is equal to "++", the real offset is the sum of the last valorised offset and the size of all the preceding fields.

Offset	Field	Type	MFD Value	SAR Value	Description
0	HEADER_TAG	E8[8]	ID=0x49,0x48 (ASCII "IH") VALID=1, VERSION=0 SIZE=48 (bytes)		Header Tag (Begin of image Leader)
8	TYPE	E8	0	1	Image type
++	SUBTYPE	E8	Radar reserved	Radar reserved	Image subtype
++	LOGCOLORS	U8	Radar reserved	0	Radar Reserved
++	IMG_RESERVED	U8	Radar Reserved		Radar Reserved
++	PRODINFO	U32[2]	Radar Reserved		Radar Reserved
++	TOD	U16[2]	Time of day		Date and time of day, as [DECD] word A1-06 and A1-07
++	RESERVED	U32	Radar Reserved		Radar Reserved
++	FCOUNTER	U32	++		Frame counter, incremented for each produced image
++	TIMETAG	TT32	Radar time-tag		Internal Radar time tag, not synchronized with 1553 time tag
++	NOMINALFRATE	U16	2500	0	Nominal frame per seconds, LSB=0.01
++	IMG_TAG	U32	Radar Reserved		Radar Reserved
++	OX	U16	0	0	Image X axis origin (pixel)
++	OY	U16	0	0	Image Y axis origin (pixel)
++	DX	U16	484	2048	Number of columns (pixel)
++	DY	U16	484	2048	Number of rows (pixel)
++	STRIDE	U16	484	2048	Number of serialized pixels per row including possible trailing, invalid pixels
++	BPP	U8	1	2	Bytes per pixel
++	COMP	U8	0		Reserved, always zero
++	SPARE	U16	0		Spare, always zero
++	PALTYPE	U8	1 (256 colors)	0 (gray scale)	Color map type: gray scale or indexed in color map palette
++	GAP	U8	0	0	Reserved, zero
56	GEO_TAG		ID=0x49,0x44 (ASCII "ID") SIZE=76		
64	INVMASK	E32	0x80000000	0x0	Bit0=1: geographic position invalid

Offset	Field	Type	MFD Value	SAR Value	Description
					Bit1=1: POI Position invalid Bit2=1: Scale invalid Bit3..30: reserved Bit31=1: POI is cursor
++	ORIENTATION	AF32	A/C true heading	Scenario dependent	Geographic orientation Y or axis
++	LATITUDE	GC32	A/C Position	Image center	Geographic latitude of pixel at [REF_Y, REF_X]
++	LONGITUDE	GC32	A/C Position	Image center	Geographic longitude of pixel at [REF_Y, REF_X]
++	REF_X	I16	241	1024	X pixel coordinate of geographic position
++	REF_Y	I16	483	1024	X pixel coordinate of geographic position
++	SCALE_X	F32	Radar Range scale meters/484	Scenario dependent	Meters per pixel
++	SCALE_Y	F32	Radar Range scale meters/484	Scenario dependent	Meters per pixel
++	POI_ORIENTATION	AF32	Radar Reserved		Radar Reserved
++	POI_LATITUDE	GC32	Cursor position	A/C position	Geographic latitude
++	POI_LONGITUDE	GC32	Cursor position	A/C Position	Geographic longitude
++	POI_X	I16	Radar Reserved		Radar Reserved
++	POI_Y	I16	Radar Reserved		Radar Reserved
++	SPARE	U32[8]	0		spare
140	RESERVED_TAG	U8[8]	ID=0x53,0x30 (ASCII "R0") SIZE=128		RES_DATA TAG
148	RES_DATA	U32[32]	Radar reserved		Radar reserved
276	CM_TAG	U8[8]	ID=0x43,0x4D (ASCII "CM") SIZE=1024		COLOUR_MAP Data
284	COLOUR_MAP	U32[256]	RGB Radar palette	Reserved	RGB palette color map
1308	PIX_TAG	BLOCK_TAG	ID=0x58,0x59 (ASCII="XY")		PIXEL Data TAG
1312	PIX_TAG.SIZE		SIZE=234256 (484*484)	SIZE= 8388608 (2024^2*2)	
1316	PIXEL_DATA	U8[PIX_TAG_SIZE]	Samples value: U8[484,484] color indexes	Samples values: U16[2048,2048] grey scale values	The image pixels

Table 4-3: Image Record detailed format

4.3.1 HEADER_DATA

The HEADER_DATA block supplies the basic information about the image dimension and pixel value interpretation:

- DX: The number of columns per row, that is the number of samples (pixel) per row.
- DY: number of rows.
- STRIDE: the serialized number of samples per row, including invalid samples trailer that could be used to align the image row to a certain number of samples. When there are not invalid trailing samples, STRIDE is equal to DX.
- BPP: number of bytes per sample.

The byte size of the image will be STRIDE*DY*BPP bytes, where for each rows the first DX*BPP are valid bytes, followed by (STRIDE-DX) * BPP trailing invalid bytes.

4.3.1.1 Frame Rate

The HEADER_DATA.FCOUNTER increments for every new image produced by the Radar, regardless it is serialized of not.

The receiver should monitor the FCOUNTER and TIMETAG values to correctly reconstruct the frame rate, where applicable (MFD only):

- The field NOMINALFRATE shall be considered a "hint":
    - Is equal zero, when the images are not directly correlated as for SAR images.
    - In equal to 2500 (/100=25fps), for the Radar MFD.
- A discontinuity of FCOUNTER indicates that some frame has been lost.
- The difference between TIMETAG of current image and TIMETAG of the previous represent the time elapsed between the two received frames.

4.3.2 PIXEL_DATA

The image pixel data associated with the HEADER_DATA shall be interpreted of a matrix of [DY, STRIDE], serialized row by rows. Pixel data can be converted in a visual element using the supplied "colour" information:

- Grey scale (PALTYPE=0). The colour map shall be ignored. The pixel value requires post-processing in accordance with the image type and BPP.
  - BPP=1: grey value from 0..255
  - BPP=2: grey value form 0..65535
- Colour map (PALTYPE=1, BPP=1): The pixel value is an index in the colour map: RGB[Y, X]=COLOUR_MAP[PIX[Y,X]]

Offset	Field	Type	Description
0	PIX[0,0]	U8/U16	Value of first upper left corner pixel, first row, first column
+BPP	PIX[0,1]	U8/U16	Value of the first rows, second column
+BPP*(DX-1)	PIX[0,DX-1]	U8/U16	Value of the first row, last column
+BPP*(STRIDE)	PIX[1,0]	U8/U16	Value of the second row, first column
…			
+BPP*(DY-1)*STRIDE+(DX-1)	PIX[DY-1][DX-1]	U8/U16	Last row, last column (last pixel) at lower right corner

4.3.3 GEO_DATA

This data block contains information for geo-referencing the image and are fully applicable only for SAR image:

- The coordinate [REF_Y, REF_X] of the georeferenced pixel. Normally equal to the centre of the image (1024,1024)
- The LATITUDE and LONGITUDE of the reference pixel
- The ORIENTATION of the Y axis of the image
- The SCALE, that is the meters per pixel

4.3.4 COLOUR_MAP

The colour map contains the conversion RGB colours applicable to image with PALTYPE=1 and BPP=1. It is composed of an array of 256 RBG values, each one represented as a U32 encoded as:

- Bit 0..7: Blue component
- Bit 8..15: Green component
- Bit 16..23: Red component
- Bit 24..31: Radar reserved

The colour map shall be always present, and it can be marked valid also for grey scale image (when it is not applicable).

5 IMAGE SERIALIZATION PROTOCOL

5.1 Ethernet interface characteristics

The Ethernet interface shall be conforming to standard IEEE 802.3ab, 1000BASE-T, Full-duplex 1Gigabit Ethernet. The IEEE 802.1Q is not supported.

The Radar shall support the following Ethernet Type frame:

Acronym	Protocol	EtherType	RFC
IPv4	Internet Protocol version 4. **IPv4 fragmentation not supported**.	0x800	RFC 791
ARP	Address Resolution Protocol	0x806	RFC 826

The Radar shall support the following IPv4 protocols:

ID	Protocol	Protocol code	RFC
ICMP	Internet Control Message Protocol: • ECHO (ICMP type 8) • ECHO Reply (ICMP type 0) • All other ICMP types: NOT SUPPORTED	1	RFC 792
UDP	User Datagram Protocol: • Options fields not supported. • Maximum UDP payload (MTU): 1460 bytes	17	RFC 768

The Radar Ethernet MAC-48 address depend on the hosting HW and Operating System.
It can be:

- The build-in MAC-48 address, where imposed by the HW or Operating System, for instance when hosted on a personal computer.
- **Factory fixed to 82:00:00:42:93:00** (unicast, locally administrated), when hosted on the Radar.

5.2 Network IPv4/UDP addressing

The network IPv4 addresses and UDP ports can be customized in accordance with the following table:

	Default	**Customizable**
IPv4 subnet	192.168.2.0/24 - **Class C address**	**Yes**, but must a **Class C address**
IPv4 Net mask	255.255.255.0 - **Mandatory**	**No**, not customizable
IPv4 unicast address	192.168.2.142	**Yes**, upon agreement. Some IP are already reserved and used by the Radar
IPv4 broadcast address	192.168.2.255	**Yes** (depend on IPv4 subnet)
Receiver UDP Port	55556	**Yes**
Receiver IPv4 address	Broadcast at start-up, unicast after reception of the first acknowledge	**Yes**, unicast, to be agreed. Some IP are already reserved and used by the Radar
Receiver MAC-48	ARP discovered	**Yes**
Unicast addressing	Broadcast at startup (receiver IPv4 and MAC-48 addresses discovery). Unicast after receiver addresses discovered	**Yes,** unicast upon IPv4 and MAC-48 receiver agreement

The above parameters can be customized upon agreement with the receiver, responsible for the design of the Avionic Network, but the subnet and related net mask **SHALL be Class C network.**

Upon agreement of the receiver:

- Fixed Receiver IPv4 address and port: the Radar shall emit only unicast IPv4 addresses, except for the Ethernet broadcast packet required for the receiver MAC-48 address to IPv4 addressing resolution (ARP protocol).
- Fixed Receiver MAC-48 address: the Radar shall not use the ARP protocol for address resolution and use only unicast MAC-48 ad IPv4 fixed addressing. If MAC-48 is fixed, then receiver IPV4 address shall be fixed.
- MFD images and SAR images can have the same receiver or two separate receivers.

5.3 Ethernet Fragmentation Protocol

The image record transmission protocol, here in after called **SFP**, is built on top of the Internet standard UDP (User Datagram Protocol).

The Radar fragments the record into multiple packets (UDP datagrams) decorated with essential information to permit reassembling and packet lost detection on the receiver site.

The Radar (the transmitter):

- "Push" the file: unsolicited and automatically begin an image transmission as soon as the image record is ready. Depending on the data origin, the transmission can be periodic or aperiodic
- It tries to transmit the data as fast as possible
- The Radar does not buffer in any way the image. If, for any reason, a new record is ready prior to the end of the previous transmission, the image is lost

A record transmission is fragmented as follow:

1. The Radar send a file "**leader**" packet (a single UDP datagram), composed of:
    - The fragmentation header, with information to uniquely identify the transaction and fragmentation information, including a special flag to inform the receiver that it will wait for an acknowledge packet
    - The first fragment of image record
2. The Radar wait for the receiver acknowledge
    - In case of timeout, the transmission is aborted, and the data record is lost
    - In case of reply, the Radar will continue with the other file fragments
3. The Radar sends "**body**" packets but the last fragment, each one carry ordered file data fragments. Each body packet is composed by:
    - The fragment header
    - The fragment body
4. The Radar sends a trailer packet, properly identified, containing the last record fragment header and last body data

Upon on receiver request, the Radar shall implement a datagram based flow control:

1. The receiver reply to the leader packet can include a request to the Radar to wait for an acknowledge packet every N datagram
2. Every N packets, the Radar shall:
    a. Send the last packet with a special flag informing the receiver that an ACK is requested
    b. Wait for receiver acknowledge. If an acknowledge packet is missed, the whole record is aborted.
3. The receiver shall monitor the ACK request sent by the Radar to send an ACK packet (changing, if desired the ACK windows)

The record fragments are not required to be of same size. The Radar is free to emit UDP datagrams of different size.

The first fragment shall always contain the entire Image Leader data from byte 0 to byte 1313. The second fragment shall begin with the value of the first pixel of the image.

5.4 Ethernet Packet Format

In accordance with IEE802 specification, the minimum Ethernet packet size is 64 bytes, while the maximum is 1514 bytes (1518, minus the trailing 4 bytes of CRC and FCS - the 8-bytes preamble is not counted).

Thus, an SFP packet over UDP is composed as follows:

Ethernet packet composition	Size	Note	Ref.
Ethernet Frame Header	14 bytes (4 bytes of 802.1Q header not supported)		IEEE 802.3ab
IPv4 Header	20 bytes (options not supported)	EtherType: 0x0800	RFC 791
UDP Header	8 bytes (options not supported)	IPv4 Protocol Code: 17	RFC 768
SFP Frag header	32 bytes		
SFP maximum payload	Up to 1428 bytes per packet		

5.5 Fragment Header format

Every datagram (fragment) sent by the Radar or by the receiver (a reply) is prefixed by a fragment header. The replayed header shall be a copy of the last received valid packet except for the contents of some fields, as explained in the following table:

Offset	Field	Type	Producer (the Radar)	Receiver reply	Notes
0	SFP_MARKER	U8	0x2A (ASCII "*")	0x2A	SFP protocol marker
1	SFP_DIRECTION	U8	0x3E (ASCII ">")	0x3C (ASCII "<")	The Transmitter (the Radar) or the consumer (reply packets)
2	SFP_VERSION	U8	0x00	As last received packet	SFP version
3	SFP_SPARE	U8	0x00	As last received packet	Spare
4	SFP_TAG	U8	0x5B: first packet ("[") 0x2B: body packet ("+") 0x5D: last packet ("]")	As last received packet	Fragment part: • Leader • Body • Trailer (last)
…	SFP_SRC	U8	0x52: Radar ("R")	As last received packet	Source ID (the Radar, also in the reply packet)
	SFP_FLOW	U8	0x4D: MFD ("M") 0x53: SAR ("S")	As last received packet	Data flow identifier
	SFP_TID	U8	0..255	As last received packet	Transaction ID
	SFP_FLAGS	E8	Bit0: ACK request	Bit0: ACK Bit7: NACK	ACK flag or NACK in case of errors
	SFP_WIN	U8	Current packet window position	Window request	0: no window requested; no ACK expected (but for the first leader fragment) 1: please ask ACK every packet 2..255: please ask ACK every SFP_WIN packets
	SFP_ERR	U8			0 = no error not equal 0: error
	SFP_ERR_INFO	U8	TBD	TBD	
	SFP_TOTFRAGS	U16	MFD=166 SAR=5876	As last received packet	Total number of frag, including the leader and trailer. If ==0, not calculated
	SFP_FRAG	U16	0… SFP_TOTFRAGS-1	As last received packet	Fragment counter
	SFP_RECTYPE	U8	0x49 (ASCII "I")	As last received packet	'0x49 = bitmap image 0 = reserved

					255 = error recovery info (change the meaning of next fields)
	SFP_RECSPARE	U8	0	As last received packet	Spare
	SFP_PLDAP	U8	0	As last received packet	Payload pad
	SFP_PLEXT	U8	0	As last received packet	Payload extension
	SFP_RECCOUNTER	U16	++	As last received packet	Record counter
	SFP_PLSIZE	U16		As last received packet	Payload size
	SFP_TOTSIZE	U32		As last received packet	Image total size
31	SFP_PLOFFSET	U32	0++	As last received packet	Payload offset: • Frag 0: always 0 • Frag N: Frag N-1 SFTP_SIZE+SFP_PLOFFSET
32	PLDATA			N/A	Payload data bytes [SFP_OFFSET..SFP_OFFSET+(SFP_PLSIZE-1)]

Table 5-1: Fragment Header format

5.6 Transmitter (the Radar)

For each new image (MFD and SAR), the Radar generate a new "transaction", associated with a unique transaction identifier (SFP_TID).

Each flow (MFD or SAR) has its private serialization "context", independent of the other flows, composed of:

- A record counter SFP_RECCOUNTER, incremented for each serialized image
- A FCOUNTER, incremented for each generated image
- An SFP_WIN, that is how make packets can be sent without waiting for receiver acknowledge:
  - Small value of SFP_WIN increase the image total transfer time
  - High value of SFP_WIN decrease the image transfer time
  - SFP_WIN=0 has special meaning: the transmitter never waits for ACK
- SFP_FRAG: increments by 1 for every fragment. The first fragment (leader) shall always have SFP_FRAG equal 0, while the last shall have SFP_FRAG= SFP_TOTFRAGS-1
- SFP_PLSIZE: valid payload size for current packet. Note that the packet could be contains trailing bytes that shall be ignored. Only the first SFP_PLSIZE bytes are to be considered
- SFP_PLOFFSET: base offset of the payload data in the reassembled image file
  - First fragment has always SFP_PLOFFSET=0
  - Next frags has always SFP_PLOFFSET=previous SFP_OFFSET + previous SFP_PLSIZE

SFP_RECCOUNTER and FCOUNTER do not necessary start from zero and are normally incremented together by one for each generated and serialized image.

5.7 Receiver

The receiver shall always acknowledge the first fragment (the leader packet) and, if it wants, can ask to the Radar to slow down transmission rate by means of the SFP_WIN field. The SFP_WIN can increase the time required to transfer an image, so it must keep in count the image transfer time and generation rate:

- MFD: should be keep as big as possible (preferably equal zero: no ACK at all), due to the high frame rate. A too small STP_WIN can cause image overrun, that is a new image is ready to be serialized prior to the end of the previous one.
- SAR: as for MFD, a too high STP_WIN can cause image overrun, but due to the low image rate, there is more time to transfer an image respect to the MFD

The receiver shall monitor the SFP_TID, SFP_RECOUNT, FCOUNTER, SFP_FRAG and SFP_TAG, to detect communication problems:

- SFP_TAG: leader without a trailer, image totally or partially lost
- SFP_FRAG not equal to previous STP_FRAG+1: fragment lost
- SFP_RECCOUNT not equal to SFP_RECCOUNT+1: image lost
- FCOUNTER not equal to previous FCOUNTER+1: image lost
  - If SFP_RECCOUNT equal to SFP_RECCOUNT+1, image overruns

The image shall be reassembled by:

- Extract the dimension (DX,DY,STRIDE,PBB) from the Leader (first frag)
- Extract the color map, if applicable
- Extract the pixel data from second frag to last payload, at required image offset

6 NOTES

6.1 Acronyms

Acronym	Meaning
MFD	Multi-Functional Display, always meaning the bitmap image of the Radar display normally visible on the Radar analogic video output
SAR	Synthetic Aperture Radar, always meaning the bitmap raw SAR image produced by the Radar.
ETH	Ethernet
IPv4	Internet Protocol version 4
UDP	User Datagram Protocol
MTU	Maximum Transmission Unit
DECD	Data Exchange Control Document
WINTEL	Wintel is the partnership of Microsoft Windows and Intel producing personal computers using Intel x86-compatible processors running Microsoft Windows. The word Wintel is a portmanteau of Windows and Intel.
FCR	Fire Control Radar (that is the GRIFO-E Radar)
SFP	Simple Fragmentation Protocol
RFC	Request for Comments, a memorandum on Internet standards
IEEE	Institute of Electrical and Electronics Engineers
MAC	Medium Access Control Address, always in the 48-bits format MAC-48 (a.k.a. EUI-48)
TX	Transmit or transmitter
RX	Receive or receiver
ACK	Acknowledge
ARP	Address Resolution Protocol
fps	Frame per seconds
Mbps	Megabits per seconds (decimal, 1M = 1000000 bits)

6.2 Wireshark SFP protocol dissector

Wireshark can be customized with a custom "protocol dissector" to examine the image serialization traffic. The dissector can be coded in LUA scripting language and installed in the Wireshark user's personal plugins directory.

The Wireshark user's personal plugins directory can be accessed using Wireshark Help dialog (Help->About Wireshark), panel "Folders", folder "Personal Lua Plugins" (double click on the location will open the folder or demand to create it if it does not exist).

6.3 Note about received image postprocessing for visualization.

6.3.1 MFD image visualization

The MFD image is essentially a 484x48 pixels, 8bits BMP like image, with color table. It can be displayed:

- as it is, using the associated color palette
- or using a customized palette to locally modify the global intensity or a per logical color intensity

In case of customized logical color conversion, the received shall consider the logical color role:

Pixel value (logical colour)	Role	Default colour	Note
0	Background	Black (RGB 0)	RGB 0
1	Background symbol occlusion	Black (RGB 0)	RGB 0
2	Category A Symbols	WHITE	WHITE=0..255, R=G=B=WHITE
3,18	Category B Symbols	WHITE	Ditto
4,5,6,16,	Category C Symbols	WHITE	Ditto
7,8,9	Category C1 Symbols	WHITE	Ditto
10,11,12	Category C2 Symbols	WHITE	Ditto
13,14,15	Category C3 Symbols	WHITE	Ditto
17..31	Reserved	Black (RGB 0)	
31..255	Raw Map	GRAY_SCALE	Linear remap from gray 0..255 to 32..255: • 32= black • 255=white (255) GRAY_SCALE=32*(Gray*7)/8

Where WHITE, as default color, means any intensity of white from 0 to 255 (always with same value of RED, GREEN and BLUE), in accordance with the maximum symbols intensity and categories relative intensity commanded to the Radar.

In case of a ground station visualization, it is not efficient to modify intensity on the transmitter side (the Radar). It is recommended to customize the palette directly on the ground station. The effect will be immediate, it does not require interaction with the Radar, and it permits more flexibility respect to the DECD interface (for instance, if desired the category A symbols could be displayed in green). The exact meaning of the categories and their symbols is out of the scope of this document.

To some categories are assigned multiple logical value. By the visualizer point of view, all the logical colour of a category shall be considered equivalent, that is always translated to the same RGB colour.

Gray scaled ground map images are displayed as background, under the Category A, B, C, C1, C2, C3 symbols, remapping the Raw map intensity from 0…255 to logical color 32..255.

The images sequence can be added to a 25fps video stream, using the desired format and compression.

6.3.2 SAR Image visualization

As for MFD, in case of a ground station visualization, it is not efficient to modify contrast or zoom on the transmitter side (the Radar). Furthermore, the Radar will not resend the image but simple adapt its visualization as background of the MFD.

On the visualizer side, the SAR image can be zoomed with a minimal or none at all loose of resolution (depending on the size of the display window), respect to the MFD. The visualizer can apply

7 ANNEXES

7.1 Annex A: MFD Image capture example

Capture of network traffic of 20 seconds of MFD images.

In the following extract of the example:

- Packet 1: begin MFD frame 7042, fragment 0 (leader) of 166
- Packet 2: receiver acknowledge (SFP_WIN=0)
- Packet 3: first body packet, first payload byte is pixel at [0,0]
- Packet 5: third body packet, first payload byte is pixel row 5, column 436
- Packets 6…166: body packets
- Packet 167: last (trailer) image packet, fragment 165 of 166 (165 + frag 0 = 166), pixels from [483,428] to [483,483]. 0.0045 seconds form leader (image transfer time)
- Packet 168: begin of next image, MFD frame 7043.
- Bold rows: receiver acknowledge

No.	Time	Src → Dst	Len	Type	Image Type	Frame#	Frag	Payload	Note
1	0.000000	55554 → 55556	1352	LEADER	MFD	7042	0/166	(484,484,1)	Image attribute
2	**0.000500**	**55556 → 55554**	**44**	**LEADER ACK**	**MFD**	**7042**	**0/166**	**WIN=0**	**Receiver ACK**
3	0.000662	55554 → 55556	1460	BODY	MFD	7042	1/166	[0,0]	First body Pixel [0,0]…
4	0.000696	55554 → 55556	1460	BODY	MFD	7042	2/166	[2,460]	Row 2, col 460
5	0.000721	55554 → 55556	1460	BODY	MFD	7042	3/166	[5,436]	Row 5, col 436
…	…	…	…	…	…	…	…	…	
162	0.004474	55554 → 55556	1460	BODY	MFD	7042	160/166	[469,56]	
163	0.004493	55554 → 55556	1460	BODY	MFD	7042	161/166	[472,32]	
164	0.004524	55554 → 55556	1460	BODY	MFD	7042	162/166	[475,8]	
165	0.004542	55554 → 55556	1460	BODY	MFD	7042	163/166	[477,468]	
166	0.004561	55554 → 55556	1460	BODY	MFD	7042	164/166	[480,444]	
167	0.004577	55554 → 55556	96	TRAILER	MFD	7042	165/166	[483,420]	Last frag Row 484, col 420
168	0.044045	55554 → 55556	1352	LEADER	MFD	7043	0/166	(484,484,1)	Next image

169	0.064542	55556 → 55554	44	LEADER ACK	MFD	7043	0/166	WIN=0	Receiver ACK

7.2 Annex B: SAR only Image capture example

Capture of network traffic of a single SAR image, without MFD interleaving images.

7.3 Annex C: SAR interleaved by MFD images capture

Capture of network traffic of a single SAR image interleaved by multiple MFD images. The ask the Radar to request an ACK every 32 packets.

This is an extract of the example:

No.	Time	Ports: Src → Dst	Len	Frag type	Image Type	Frame	Frags	Payload	Notes
1	0.000000	55555 → 55556	1352	LEADER	SAR	20	0/5876	(2048,2048,2)	Image attribute
2	0.001747	55554 → 55556	1352	LEADER	MFD	1881	0/166	(484,484,1)	Image attribute
3	**0.002636**	**55556 → 55555**	**44**	**LEADER ACK**	**SAR**	**20**	**0/5876**	**WIN=32**	**SAR ACK Packet 1**
4	0.002702	55555 → 55556	1460	BODY	SAR	20	1/5876	[0,0]	
5	0.002741	55555 → 55556	1460	BODY	SAR	20	2/5876	[0,714]	
6	**0.002753**	**55556 → 55554**	**44**	**LEADER ACK**	**MFD**	**1881**	**0/166**	**WIN=0**	**MFD ACK Packet 2**
7	0.002783	55555 → 55556	1460	BODY	SAR	20	3/5876	[0,1428]	
8	0.002798	55554 → 55556	1460	BODY	MFD	1881	1/166	[0,0]	
9	0.002842	55554 → 55556	1460	BODY	MFD	1881	2/166	[2,460]	
10	0.002847	55554 → 55556	1460	BODY	MFD	1881	3/166	[5,436]	
11	0.002867	55554 → 55556	1460	BODY	MFD	1881	4/166	[8,412]	
12	0.002872	55554 → 55556	1460	BODY	MFD	1881	5/166	[11,388]	
13	0.002875	55555 → 55556	1460	BODY	SAR	20	4/5876	[1,94]	
14	0.002878	55554 → 55556	1460	BODY	MFD	1881	6/166	[14,364]	
15	0.002910	55555 → 55556	1460	BODY	SAR	20	5/5876	[1,808]	
16	0.002915	55554 → 55556	1460	BODY	MFD	1881	7/166	[17,340]	
17	0.002919	55555 → 55556	1460	BODY	SAR	20	6/5876	[1,1522]	

18	0.002922	55554 → 55556	1460	BODY	MFD	1881	8/166	[20,316]	
19	0.002926	55555 → 55556	1460	BODY	SAR	20	7/5876	[2,188]	
20	0.002929	55554 → 55556	1460	BODY	MFD	1881	9/166	[23,292]	
21	0.002932	55555 → 55556	1460	BODY	SAR	20	8/5876	[2,902]	
22	0.002934	55554 → 55556	1460	BODY	MFD	1881	10/166	[26,268]	
23	0.002938	55555 → 55556	1460	BODY	SAR	20	9/5876	[2,1616]	
24	0.002941	55554 → 55556	1460	BODY	MFD	1881	11/166	[29,244]	
...									
68	0.003294	55554 → 55556	1460	BODY	MFD	1881	34/166	[97,176]	
69	0.003296	55555 → 55556	1460	BODY	SAR	20	31/5876	[10,940]	
70	0.003300	55554 → 55556	1460	BODY	MFD	1881	35/166	[100,152]	
71	0.003320	55555 → 55556	1460	BODY	SAR	20	32/5876	[10,1654]	SAR ACK Request
72	0.003324	55554 → 55556	1460	BODY	MFD	1881	36/166	[103,128]	
73	0.003329	55554 → 55556	1460	BODY	MFD	1881	37/166	[106,104]	
74	0.003340	55554 → 55556	1460	BODY	MFD	1881	38/166	[109,80]	
75	0.003343	55554 → 55556	1460	BODY	MFD	1881	39/166	[112,56]	
76	0.003485	55554 → 55556	1460	BODY	MFD	1881	40/166	[115,32]	
...									
101	0.003606	55554 → 55556	1460	BODY	MFD	1881	65/166	[188,400]	
102	0.003609	55554 → 55556	1460	BODY	MFD	1881	66/166	[191,376]	
103	**0.003615**	**55556 → 55555**	**44**	**BODY ACK**	**SAR**	**20**	**32/5876**	**WIN=32**	**SAR ack Packet 71**
104	0.003771	55555 → 55556	1460	BODY	SAR	20	33/5876	[11,320]	

105	0.003796	55555 → 55556	1460	BODY	SAR	20	34/5876	[11,1034]	
106	0.003816	55555 → 55556	1460	BODY	SAR	20	35/5876	[11,1748]	
..									
290	0.005909	55554 → 55556	1460	BODY	MFD	1881	155/166	[454,176]	
291	0.005912	55554 → 55556	1460	BODY	MFD	1881	156/166	[457,152]	
292	**0.005915**	**55556 → 55555**	**44**	**BODY ACL**	**SAR**	**20**	**128/5876**	**WIN=32**	**SAR ACK frag 128**
293	0.006021	55554 → 55556	1460	BODY	MFD	1881	157/166	[460,128]	
294	0.006053	55555 → 55556	1460	BODY	SAR	20	129/5876	[44,1280]	
295	0.006073	55554 → 55556	1460	BODY	MFD	1881	158/166	[463,104]	
296	0.006095	55555 → 55556	1460	BODY	SAR	20	130/5876	[44,1994]	
297	0.006114	55554 → 55556	1460	BODY	MFD	1881	159/166	[466,80]	
298	0.006140	55554 → 55556	1460	BODY	MFD	1881	160/166	[469,56]	
299	0.006159	55554 → 55556	1460	BODY	MFD	1881	161/166	[472,32]	
300	0.006163	55555 → 55556	1460	BODY	SAR	20	131/5876	[45,660]	
301	0.006167	55554 → 55556	1460	BODY	MFD	1881	162/166	[475,8]	
302	0.006170	55555 → 55556	1460	BODY	SAR	20	132/5876	[45,1374]	
303	0.006173	55554 → 55556	1460	BODY	MFD	1881	163/166	[477,468]	
304	0.006229	55555 → 55556	1460	BODY	SAR	20	133/5876	[46,40]	
305	0.006241	55554 → 55556	1460	BODY	MFD	1881	164/166	[480,444]	
306	0.006243	55555 → 55556	1460	BODY	SAR	20	134/5876	[46,754]	
307	0.006246	55554 → 55556	96	TRAILER	MFD	1881	165/166	[483,420]	End MFD
308	0.006248	55555 → 55556	1460	BODY	SAR	20	135/5876	[46,1468]	

...									
6887	0.189571	55555 → 55556	1460	BODY	SAR	20	5867/5876	[2045,164]	
6888	0.189592	55555 → 55556	1460	BODY	SAR	20	5868/5876	[2045,878]	
6889	0.189614	55555 → 55556	1460	BODY	SAR	20	5869/5876	[2045,1592]	
6890	0.189639	55555 → 55556	1460	BODY	SAR	20	5870/5876	[2046,258]	
6891	0.189664	55555 → 55556	1460	BODY	SAR	20	5871/5876	[2046,972]	
6892	0.189684	55555 → 55556	1460	BODY	SAR	20	5872/5876	[2046,1686]	
6893	0.189705	55555 → 55556	1460	BODY	SAR	20	5873/5876	[2047,352]	
6894	0.189723	55555 → 55556	1460	BODY	SAR	20	5874/5876	[2047,1066]	
6895	0.189766	55555 → 55556	568	TRAILER	SAR	20	5875/5876	[2047,1780]	End SAR

7.4 Annex D: Wireshark dissector

File grifo_image_protocol_dissector.lua: A trivial and essential Wireshark dissector written in LUA

