MISB – MOTION IMAGERY STANDARDS BOARD **STANDARD** **UAS Datalink Local Set**	**MISB ST 0601.8** **23 October 2014**

1 Scope

MISB Standard 0601 details the Unmanned Air System (UAS) Datalink Local Set (LS) for UAS platforms. The UAS Datalink LS is an extensible SMPTE (Society of Motion Picture Television Engineers) Key-Length-Value (KLV) metadata set designed for transmission through a wireless communications link (Datalink).

This Standard provides direction and requirements for the creation of a SMPTE ST 336 compliant Local Set (LS) for a reliable, bandwidth-efficient exchange of metadata among digital Motion Imagery systems. This Standard also provides a mapping to Predator Exploitation Support Data (ESD) for continued support of existing metadata systems.

The UAS Local Set is intended to be produced locally within a UAS airborne platform and included in an MPEG-2 Transport Stream (or equivalent transport mechanism). The MPEG-2 Transport Stream (or equivalent) also contains compressed Motion Imagery from sensors, such as Electro-Optical / Infrared (EO/IR). Synchronization between the metadata and the appropriate Motion Imagery is highly desired and is the responsibility of the system designer. The MPEG-2 Transport Stream (or equivalent) embedded with a UAS LS is transmitted over a medium bandwidth (e.g. 1 to 5Mb/s) wireless Datalink for dissemination.

The scope of this document is to provide a framework for an extensible bandwidth-efficient Local Set that enhances sensor-captured imagery with relevant metadata. This Standard also provides a mapping between UAS Datalink Local Set items, ESD items, and Universal Set (US) items defined in the latest SMPTE KLV dictionary (RP 210) as well as in the MISB-managed ST 0807 keyspace.

2 References

2.1 Normative References

The following references and the references contained therein are normative.

[1] SMPTE ST 336:2007, Data Encoding Protocol Using Key-Length-Value
[2] MISB EG 0104.5, Predator UAV Basic Universal Metadata Set, Dec 2006
[3] SMPTE RP 210v13:2012, Metadata Element Dictionary
[4] MISB ST 0603.2 Common Time Reference for Digital Motion Imagery Using Coordinated Universal Time (UTC), Feb 2014
[5] MISB ST 0604.3 Time Stamping Compressed Motion Imagery, Feb 2014

[6] MISB ST 1402 MPEG-2 Transport of Compressed Motion Imagery and Metadata, Feb 2014
[7] MISB ST 0605.4 Time Stamping and Metadata Transport in High Definition Uncompressed Motion Imagery, Feb 2014
[8] MISB ST 0107.2 Bit and Byte Order for Metadata in Motion Imagery Files and Streams, Feb 2014
[9] MISB ST 0807.13 MISB KLV Metadata Dictionary, Feb 2014
[10] MISB ST 0806.4 Remote Video Terminal Local set, Feb 2014
[11] MISB ST 0102.10 Security Metadata Universal and Local Sets for Digital Motion Imagery, Oct 2013
[12] MISB ST 0902.3 Motion Imagery Sensor Minimum Metadata Set, Feb 2014
[13] MISB RP 0903 Video Moving Target Indicator Local Set, Sep 2009
[14] ASI-00209 Rev D, Exploitation Support Data (ESD) External Interface Control Document, 04 Dec, 2002
[15] IEEE 1003.1-2013, Information Technology – Portable Operating System Interface (POSIX) Base Specifications, Issue 7
[16] ISO/IEC 646:1991, Information Technology – ISO 7-bit coded Character Set for Information Interchange
[17] MIL-STD-2500C, National Imagery Transmission Format Version 2.1, May 2006
[18] MISB ST 1204.1 Motion Imagery Identification System (MIIS) Core Identifier, Oct 2013
[19] MISB ST 1206 SAR Motion Imagery Metadata, Feb 2014

2.2 Informative References

[20] MISB ST 0806.4 Remote Video Terminal Local Set, Feb 2014
[21] MISB ST 0801.5 Photogrammetry Metadata Set for Digital Motion Imagery, Feb 2014

3 Acronyms

BER Basic Encoding Rules
ESD Exploitation Support Data
KLV Key Length Value
LS Local Set
MI Motion Imagery
OID Object IDentifer
US Universal Set

4 Revision History

Revision	Date	Summary of Changes
ST 0601.8	6/18/2014	• EARS requirements format and other general formatting • Changed LDS to LS and UDS to US

5 Introduction

A SMPTE ST 336 [1] Universal Set (US) provides access to a range of KLV formatted metadata items. Transmitting the 16-byte key, basic encoding rules (BER) formatted length, and data value is appropriate for applications where bandwidth isn't a concern. However, transmitting the 16-byte universal key consumes precious bandwidth in bandwidth-challenged environments.

The Motion Imagery Standards Board (MISB) Engineering Guideline MISB EG 0104.5 [2] entitled "Predator UAV Basic Universal Metadata Set" shows a translation between basic ESD and Universal Set (US) metadata items that exist in the most current version of the SMPTE KLV dictionary. The US items in EG 0104.5 are more appropriate for higher bandwidth interfaces (e.g. > 10Mb/s), such as dissemination, whereas this document targets low to medium bandwidth interfaces (e.g. 1 to 5Mb/s).

UAS airborne platforms typically use a wireless communications channel that allots a limited amount of bandwidth for metadata. Because of the bandwidth disadvantages in using a Universal Set, it is more desirable to use a Local Set construction for transmission over a UAS Datalink. As discussed in SMPTE ST 336, a Local Set can use a 1, 2 or 4-byte tag with a 1, 2, 4-byte, or BER (Basic Encoding Rules) encoded length. The UAS Local Set described in this document uses BER-encoded lengths and BER-OID encoded tags to minimize bandwidth requirements, while still allowing the LS ample room for growth.

This Standard identifies a way to encode metadata into a standard KLV Local Set. This standardized method is intended to be extensible to include future relevant metadata with mappings between new LS, US, and ESD (Exploitation Support Data) metadata items (where appropriate). When a new metadata LS item is added or required, the item will be added to the to the proper metadata dictionary (public or private), if the metadata item does not already exist.

The method described in this Standard also provides a mapping between Local Set items and currently implemented Universal Set items defined in the SMPTE RP 210 [3] KLV dictionary.

5.1 Local Set Changes and Updates

This document defines the UAS Datalink Metadata Local Set and is under configuration management.

Requirement	
ST 0601.8-01	Any changes to MISB ST 0601 shall be accompanied by a document revision and date change and coordinated with the managing organization.
ST 0601.8-02	Applications that implement MISB ST 0601 shall allow for metadata items in the UAS Local Set that are unknown so that they are forward compatible with future versions of the interface.

6 UAS Datalink Local Set - Requirements

These requirements for the UAS Datalink Local Set (LS) are outlined here and used as references from within this text.

6.1 KLV Rules

Requirement	
ST 0601.8-03	All UAS Datalink metadata shall be expressed in accordance with MISB ST 0107[8].
ST 0601.8-04	All UAS Datalink metadata shall be formatted in compliance with SMPTE ST 336 [1].
ST 0601.8-05	Implementations of MISB ST 0601 shall parse unknown but properly formatted metadata UAS Local Set packets so as to not impact the reading of known Tags within the same instance.
ST 0601.8-06	All instances of item Tags within a UAS Datalink LS packet shall be BER-OID encoded using the fewest possible bytes in accordance with SMPTE ST 336 [1].
ST 0601.8-07	All instances of item length fields within a UAS Datalink LS packet shall be BER Short or BER Long encoded using the fewest possible bytes in accordance with SMPTE ST 336 [1].
ST 0601.8-08	All instances of a UAS Datalink LS where the computed checksum is not identical to the included checksum shall be discarded.

6.2 Mandatory UAS Datalink LS items

Requirement	
ST 0601.8-09	All instances of a UAS Datalink LS shall contain as their first element Tag 2, UNIX Time Stamp – Microseconds.
ST 0601.8-10	The value assigned to the UNIX Time Stamp - Microseconds item (Tag 2) shall represent the time of birth of the metadata of all the elements contained in that instance of the UAS Datalink LS.
ST 0601.8-11	All instances of the UAS Datalink LS shall contain as the final element Tag 1, (Checksum).
ST 0601.8-12	All instances of the UAS Datalink LS shall contain Tag 65, UAS LS Version Number.

6.3 Metadata Usage

Requirement	
ST 0601.8-13	Excepting the requirements for Tag 2 at the start and Tag 1 at the end of a UAS Datalink LS any instance of the UAS LS, an instance of an UAS LS containing any number of properly formatted, unique Tags in any order shall be valid.
ST 0601.8-14	The usage of all Tags within the UAS Datalink LS shall be consistent with the descriptions and clarifications contained within MISB ST 0601.
ST 0601.8-15	UAS Datalink LS elements that have incomplete descriptions (i.e.: “TBD”) shall be informative rather than normative.
ST 0601.8-16	UAS Datalink LS decoding systems that understand the full-range representation of certain metadata items shall use the full-range representation and ignore the

	range-restricted representation when both exist in the same UAS Datalink LS packet.
ST 0601.8-17	UAS Datalink LS decoding systems that understand the Height Above Ellipsoid (HAE) representation of certain metadata items shall use the HAE representation and ignore the Mean Sea Level (MSL) representation when both exist in the same UAS Datalink LS packet.

6.4 LS Universal Keys

Requirement	
ST 0601.8-18	The UAS Local Set 16-byte Universal Key shall be 06 0E 2B 34 - 02 0B 01 01 – 0E 01 03 01 - 01 00 00 00 (CRC 56773)

UAS Datalink LS Universal Key history

Date Released: May 2006

Description: Defined in MISB ST 0807 [9]

A key history is provided below as a way to track the keys used in engineering and development. Note that the keys listed below are informative only.

DO NOT use the below historical universal keys in any future development.
Key: `06 0E 2B 34 - 01 01 01 01 - 0F 00 00 00 - 00 00 00 00` ***Date Released:*** November 2005 ***Description:*** Experimental node key used in software development efforts at General Atomics prior to the assignment of a defined key.
Key: `06 0E 2B 34 - 02 03 01 01 - 01 79 01 01 - 01 xx xx xx` ***Date Released:*** October 25, 2005 ***Description:*** This key was released as a placeholder within early versions this document. Much development has been based around draft versions of this document which has used this key in some software implementations.

Requirement	
ST 0601.8-19	Historical 16-byte Universal Keys shall not be used in future developments.

6.4.1 SMPTE Universal Key Version Number

Depreciated in ST 0601.6.

6.5 LS Packet Structure

Figure 6-1 shows the general format of how the LS is configured. It is required that each LS packet contain a Precision Timestamp (defined in MISB ST 0603 [4]), which is Coordinated

Universal Time (UTC) - based that represents the time of birth of the metadata within the LS packet to conform with the requirements in Section 6.2. Time stamping is further discussed in Section 6.7. A checksum metadata item is also required to be included in each LS packet and needs to conform with the requirements in Section 6.2. Checksums are discussed in Section 6.8.

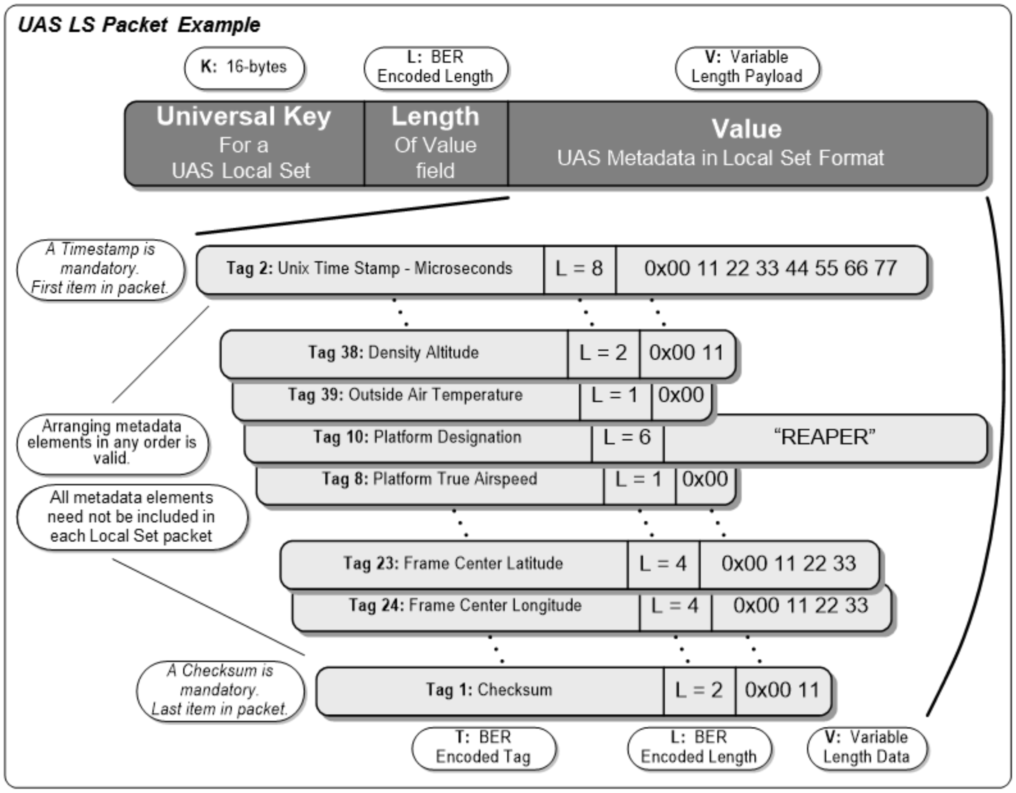

Figure 6-1: Example of a UAV Local Set Packet

Any combination of metadata items can be included in a UAS Local Set packet. Also, the items within the UAV LS can be arranged in any order. However, the timestamp is always positioned at the beginning of an LS packet, and the checksum always appears as the last metadata item, which aids algorithms surrounding its computation and creation (see requirements in Section 6.2).

6.5.1 Bit and Byte Ordering

All metadata is represented using big-endian (Most Significant Byte (MSB) first) encoding, and Bytes using big-endian bit encoding (most significant bit (msb) first) (see requirements in Section 6.1).

6.5.2 Tag and Length Field Encoding

The UAS LS item tag and length fields are encoded using basic encoding rules (BER) for either short or long form encoding of octets (see requirements in Section 6.1). This length encoding method provides the greatest level of flexibility for variable length data contained within a KLV packet.

See SMPTE ST 336 for further details.

6.5.2.1 BER Short Form Length Encoding Example

For UAS LS packets and data elements shorter than 128 bytes, the length field is encoded using the BER short form. Length field using the short form are represented using a single byte (8 bits). The most significant bit in this byte signals that the long form is being used. The last seven bits depict the number of bytes that follow the BER encoded length. An example LS packet using a short form encoded length is shown in Figure 6-2:

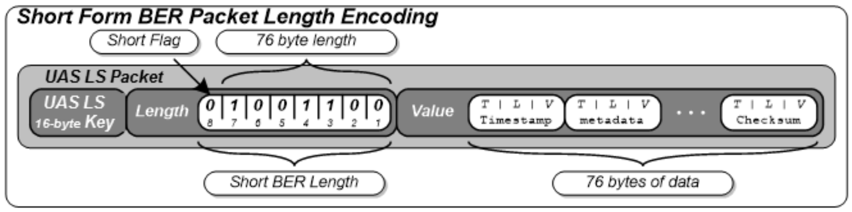

Figure 6-2: Example Short Form Length Encoding

Although this example depicts the length field of the entire LS packet, short form BER encoding also applies to the individual item lengths within the LS packet.

6.5.2.2 BER Long Form Length Encoding

For UAS LS packets and data elements longer than 127 bytes, the length field is encoded using the BER long form. The long form encodes length field using multiple bytes. The first byte indicates long form encoding as well as the number of subsequent bytes that represent the length. The bytes that follow the leading byte are the encoding of an unsigned binary integer equal to the number of bytes in the packet. An example LS packet using a long form encoded length is shown in Figure 6-3:

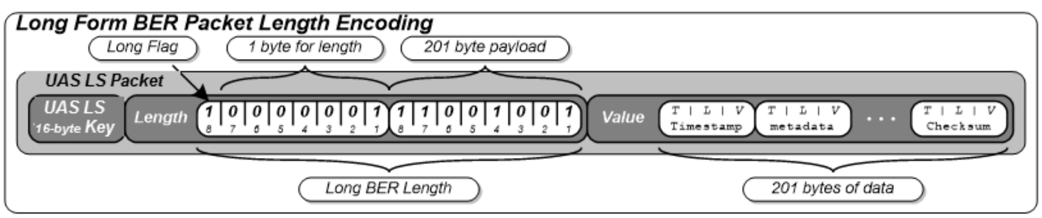

Figure 6-3: Example Long Form Length Encoding

Although this example depicts long form BER encoding on the length field of the entire LS packet, long form BER encoding also applies to the individual item lengths within the LS packet.

6.5.2.3 BER-OID Encoding for Tags

Also known as “primitive BER”, or “ASN.1 OID BER”, BER-OID encoding of tags differs from short and long form BER encoding used for KLV lengths as described in Sections 6.5.2.1 and 6.5.2.2.

Local KLV sets employing the use of BER-OID encoded tags can represent an almost limitless number of metadata items. BER-OID binary encoding allows the size of a tag space to increase through the inclusion of additional bytes when the tag number passes certain threshold.

For instance, one BER-OID byte (or octet) can represent up to 127 different metadata items. Two bytes can represent 16,383 items. Generalizing for any number of bytes "N" used as a BER-OID tag, the number of tags that can be represented is $2^{7 \cdot N} - 1$.

When using BER-OID encoding for tags, each tag is represented as a series of one or more bytes. Bit 8 (msb) of each byte indicates whether it is the last in the series: bit 8 of the last byte (LSB) is zero, while bit 8 of each preceding byte (MSB's) is one. Bits 7 to 1 of the bytes in the series collectively encode the metadata tag.

Conceptually, these groups of bits are concatenated to form an unsigned binary number whose most significant bit is bit 7 of the first bite, and whose lease significant bit is bit 1 of the last octet.

A BER-OID encoded tag must use the fewest bytes possible. Equivalently, the leading byte(s) of the BER-OID tag must not have the value of 0x80.

BER-OID encoding examples for one, two, and three-byte encodings are shown in Figure 6-4, Figure 6-5 and Figure 6-6 respectively.

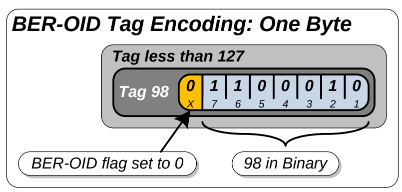

Figure 6-4: BER-OID Tag Encoding Using One Byte

Note that only 127 different tags are encoded using a single byte. Decoding is the reverse of encoding. This is the only tag encoding currently encountered in the UAS LS.

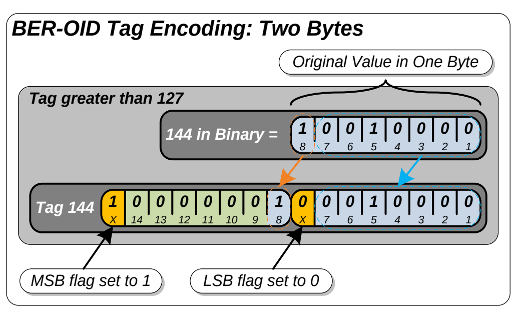

Figure 6-5: BER-OID Tag Encoding Using Two Bytes

Note that logical tags 128 through 16,383 are encoded using two bytes. Decoding is the reverse of encoding.

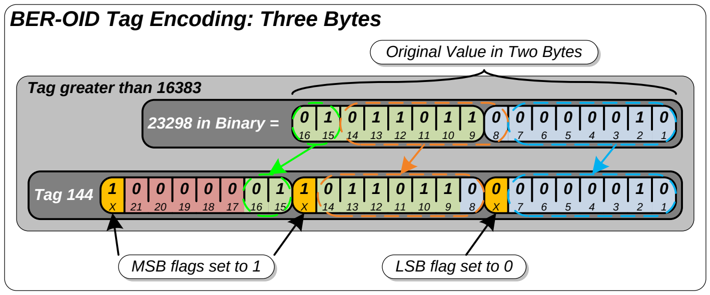

Figure 6-6: BER-OID Tag Encoding Using Three Bytes

Note that logical tags 16,384 through 2,097,151 are encoded using three bytes. Decoding is the reverse of encoding.

Although not currently in use, it is envisioned that a maximum of 2-bytes will be used to encode BER-OID tags within the UAS LS in future revisions.

6.5.3 Nesting Local Sets within the UAS Datalink LS

To provide a method to re-use commonly used metadata field from the UAS LS (platform location, and sensor pointing angles) while providing greater flexibility to system implementers, other Local Sets (with tag defined in the UAS LS) may be nested within the UAS LS.

A nested Local Set is treated the same as any other standalone metadata item defined within the UAS LS where the Tag is defined by this document, and the length field is determined by the size of the value portion. The nested set, however, typically has an increased length compared to other UAS LS items, and the value portion conforms to the defining Local Set document. An illustrative example packet showing the RVT LS (MISB ST 0806 [20]) nested within the UAS LS is shown in Figure 6-7.

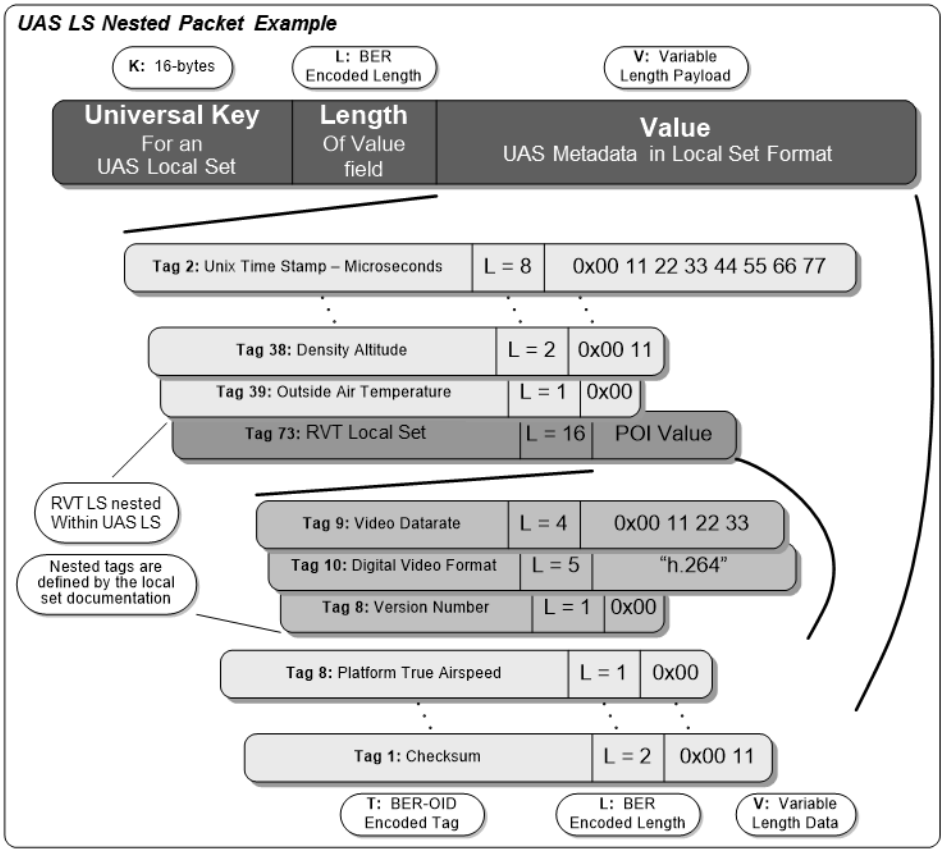

Figure 6-7: Nested Packet Example

6.6 Data Collection and Dissemination – Informative

Within the air vehicle, metadata is collected, processed, and then distributed by the flight computer (or equivalent) through the most appropriate interface (SMPTE Serial Digital Interface (SDI), RS-422, 1553, Ethernet, Firewire, etc.). See Figure 6-8.

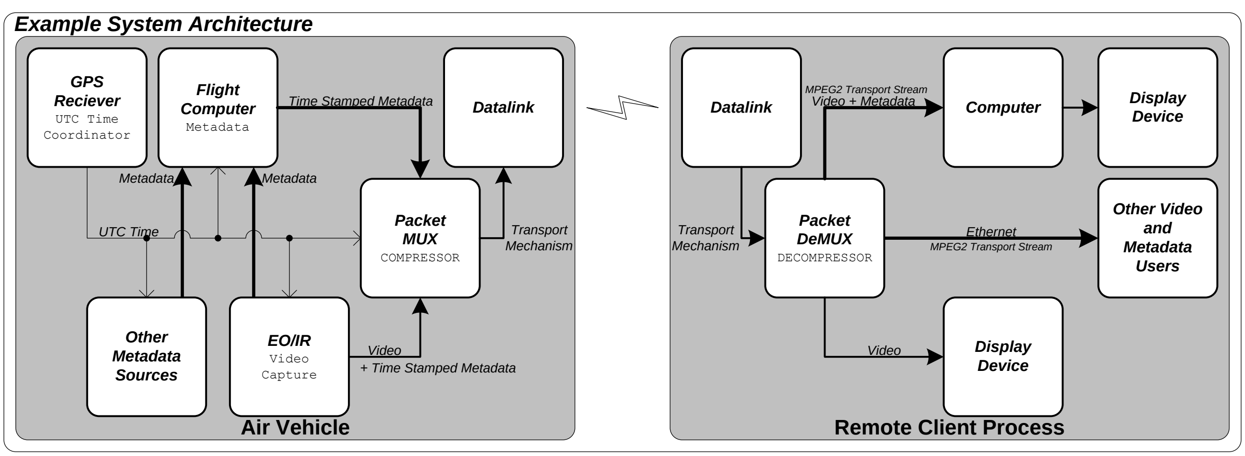

Figure 6-8: System Architecture

Sensors and other metadata sources pass metadata to the flight computer.

The flight computer (or equivalent) places a timestamp in the UAS LS packet prior to passing it to the Video Encoder / Packet Multiplexer. See Section 6.7 for more information about using timestamps in the LS metadata packet.

Although the means for packaging Motion Imagery with metadata may be implementation specific, the following provides a general idaea of the process. The flight computer merges all appropriate metadata items along with a timestamp and checksum into a LS packet and sends the data to a Motion Imagery encoder/packet multiplexer, which produces a unified data stream for off-platfrom transmission. Once passed through the communications link, a remote client can decode and process the Motion Imagery and metadata contained within the stream. Users can then display and/or distribute the Motion Imagery and metadata as appropriate.

6.7 Time Stamping

Every LS KLV packet is required to include a Precision Time Stamp as defined in MISB ST 0603 as a way to correspond the metadata with a standardized time reference. The Precision Time Stamp is based on UTC time, which provides a means to associate metadata with Motion Imagery frames, and for reviewing time-critical events at a later date. This section describes how to include the mandatory timestamp within a UAS Local Set packet according to the requiements in Section 6.2

Metadata sources and the flight computer (or equivalent) are coordinated to operate on the same standard time, which is typically GPS derived. The metadata source provides a timestamp for inclusion in a LS packet and the timestamp assists the accuracy of synchronizing each frame to its corresponding metadata set.

The timestamp (Tag 2) is an 8 byte unsigned integer that represents the number of microseconds that have elapsed since midnight (00:00:00), January 1, 1970. This date is known as the Unix epoch (POSIX Microseconds) and is discussed in the IEEE POSIX standard IEEE 1003.1.

When receiving packets of ST 0601 metadata, the time value represents the time of birth of all metadata items contained within the UAS LS packet in accordance with the reqirements in Section 6.2. When generating UAS LS metadata packets, the most current metadata samples since the last metadata packet (with timestamp) are intended to be used and assigned the current time.

Generation of metadata packets introduces a situation where the time of birth timestamp may not directly correspond to when a metadata value was actually sampled. In this case, the maximum timestamp error encountered is the difference in time between the current metadata packet, and the packet which immediately precedes it.

Systems producing or accepting ST 0601 metadata streams are allowed to adjust metadata repetition rates to meet timestamp precision needs.

6.7.1 Packet Timestamp

An LS Packet Timestamp is inserted at the beginning of the value portion of a UAS LS packet.

The UTC timestamp represented by Tag 2 (UNIX Timestamp) applies to all metadata in the LS packet. This timestamp corresponds to the time of birth of all the data within the LS packet. This time can be used to associate the metadata with a particular video frame and be displayed or monitored appropriately.

An example LS packet containing a timestamp is show in Figure 6-9:

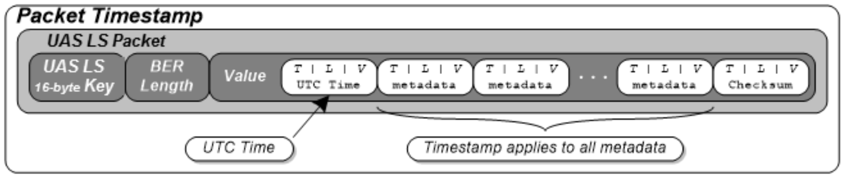

Figure 6-9: Packet Timestamp Example

6.8 Error Detection

To help prevent erroneous metadata from being presented with the Motion Imagery, it is required that a 16-bit checksum is included in every UAS Local Set packet as the last item (see requirements in Section 6.2). The checksum is a running 16-bit sum through the entire LS packet starting with the 16 byte Local Set key and ending with summing the length field of the checksum data item.

Figure 6-10 shows the data range that the checksum is performed over:

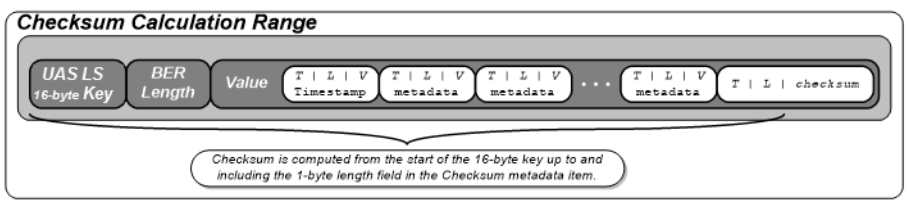

Figure 6-10: Checksum Computation Range

An example algorithm for calculating the checksum is shown below:

```
unsigned short bcc_16 (
  unsigned char * buff,  // Pointer to the first byte in the 16-byte UAS LS key.
  unsigned short len )   // Length from 16-byte US key up to 1-byte checksum length.
{
  // Initialize Checksum and counter variables.
  unsigned short bcc = 0, i;

  // Sum each 16-bit chunk within the buffer into a checksum
  for ( i = 0 ; i < len; i++)
    bcc += buff[i] << (8 * ((i + 1) % 2));
  return bcc;
}  // end of bcc_16 ()
```

If the calculated checksum of the received LS packet does not match the checksum stored in the packet, the user must discard this packet as being invalid (see requirements in Section 6.1). The lost LS packet is of little concern since another packet is available within reasonable proximity (in both data and time) to this lost packet.

6.9 Motion Imagery/Metadata Synchronization

Synchronization or time-alignment of a Motion Imagery frame with metadata is highly desired and is the responsibility of the system designer. The Precision Time Stamp, referenced in this document, is based on UTC and the POSIX Epoch; requirements for its use is outlined in MISB ST 0603 [4]. Requirements for time stamping compressed Motion Imagery with a Precision Time Stamp are outlined in MISB ST 0604 [5]. Methods and requirements for synchronizing compressed Motion Imagery and metadata within an MPEG-2 Transport Stream are discussed in MISB ST 1402 [6]. Requirements for time stamping and metadata carriage in high definition uncompressed Motion Imagery are outlined in MISB ST 0605 [7].

Many considerations need to be weighed in specifying the intent in synchronizing Motion Imagery frames with metadata. These include: sufficient bandwidth to accommodate the metadata without limiting the Motion Imagery; required update rates of metadata; requirements for presentation of synchronized Motion Imagery with metadata at a client receiver; receiver decoder buffer (delay) requirements. Different applications will have differing requirements on how tight the synchronization needs to be, and whether sufficient information is available to guarantee the relationship between the Motion Imagery and the metadata. While metrics for the timing of Motion Imagery and metadata may be application specific, the best advice at this time is to ensure that the Precision Time Stamp when inserted into a Motion Imagery frame and into a metadata local set is as accurate to the point of collection s possible for both.

7 UAS Local Set Tables

This section defines the content of the UAS Local set as well as translation between LS & ESD, and LS and US data types.

For guidance on which items to include in ST 0601 packets, refer to ST 0902 (Motion Imagery Sensor Minimum Metadata Set) for a listing of a minimum set of UAS LS metadata items.

7.1 UAS DataLink Local Set Items

Each UAS Local Set item is assigned an integer value for its tag, a descriptive name, and also has fields indicating the units, range, format, and length of the data item. More detailed information about the data item is included in the Notes column.

Notes:

- The columns labeled “Mapped US”, “Units”, “Format”, “Len” (for length) and “Notes” all apply to the Local Set ONLY and not ESD or US data types.
- “ESD Name” is the name assigned to an ESD metadata item labeled as a two-character digraph in the “ESD” column.
- An “x” within a field below indicates that no data is available.

- The “Mapped US” column is the Universall set metadata key reserved to represent the length and data format specified by the referring LS metadata item. The key is the only parameter which differs between US and tag of the LS item. Note that LS items which state “Use EG 0104 US Key” may require conversion between LS and US data types prior to representing an LS item as a US item.
- The “US” column is an existing metadata key which the UAS LS is mapped to in some applications (i.e.: EG 0104). Note that the LS and EG 0104 data formats often differ between one another and a US key could not be used to represent the data in an LS item without proper conversion first.

Table 1: UAS Datalink Metadata Set

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
1	Checksum	06 0E 2B 34 01 01 01 01 0E 01 02 03 01 00 00 00 (CRC 56132)	x	x	x	x	None	uint16	2	Checksum used to detect errors within a UAV Local Set packet. Lower 16-bits of summation. Performed on entire LS packet, including 16-byte US key and 1-byte checksum length.
2	UNIX Time Stamp	Use EG0104 US Key	x	x	06 0E 2B 34 01 01 01 03 07 02 01 01 01 05 00 00 (CRC 64827)	User Defined Time Stamp - microseconds since 1970	Microseconds	uint64	8	Coordinated Universal Time (UTC) represented in the number of microseconds elapsed since midnight (00:00:00), January 1, 1970. Derived from the POSIX IEEE 1003.1 standard. Resolution: 1 microsecond.
3	Mission ID	06 0E 2B 34 01 01 01 01 0E 01 04 01 03 00 00 00 (CRC 65358)	Mn	Mission Number	06 0E 2B 34 01 01 01 01 01 05 05 00 00 00 00 00 (CRC 37735)	Episode Number	String	ISO 646	V	Descriptive Mission Identifier to distinguish event or sortie. Value field is Free Text. Maximum 127 characters.
4	Platform Tail Number	06 0E 2B 34 01 01 01 01 0E 01 04 01 02 00 00 00 (CRC 35322)	Pt	Platform Tail Number	x	x	String	ISO 646	V	Identifier of platform as posted. E.g.: "AF008", "BP101", etc. Value field is Free Text. Maximum 127 characters.
5	Platform Heading Angle	Use EG0104 US Key	Ih	UAV Heading (INS)	06 0E 2B 34 01 01 01 07 07 01 10 01 06 00 00 00 (CRC 23727)	Platform Heading Angle	Degrees	uint16	2	Aircraft heading angle. Relative between longitudinal axis and True North measured in the horizontal plane. Map 0..(2^16-1) to 0..360. Resolution: ~5.5 milli degrees.
6	Platform Pitch Angle	Use EG0104 US Key	Ip	UAV Pitch (INS)	06 0E 2B 34 01 01 01 07 07 01 10 01 05 00 00 00 (CRC 51059)	Platform Pitch Angle	Degrees	int16	2	Aircraft pitch angle. Angle between longitudinal axis and horizontal plane. Positive angles above horizontal plane. Map -(2^15-1)..(2^15-1) to +/-20. Use -(2^15) as "out of range" indicator. -(2^15) = 0x8000. Resolution: ~610 micro degrees.
7	Platform Roll Angle	Use EG0104 US Key	Ir	UAV Roll (INS)	06 0E 2B 34 01 01 01 07 07 01 10 01 04 00 00 00 (CRC 45511)	Platform Roll Angle	Degrees	int16	2	Platform roll angle. Angle between transverse axis and transvers-longitudinal plane. Positive angles for lowered right wing. Map (-2^15-1)..(2^15-1) to +/-50. Use -(2^15) as "out of range" indicator. -(2^15) = 0x8000. Res: ~1525 micro deg.
8	Platform True Airspeed	06 0E 2B 34 01 01 01 01 0E 01 01 01 0A 00 00 00 (CRC 20280)	As	True Airspeed	x	x	Meters/Second	uint8	1	True airspeed (TAS) of platform. Indicated Airspeed adjusted for temperature and altitude. 0..255 meters/sec. 1 m/s = 1.94384449 knots. Resolution: 1 meter/second.
9	Platform Indicated Airspeed	06 0E 2B 34 01 01 01 01 0E 01 01 01 0B 00 00 00 (CRC 14732)	Ai	Indicated Airspeed	x	x	Meters/Second	uint8	1	Indicated airspeed (IAS) of platform. Derived from Pitot tube and static pressure sensors. 0..255 meters/sec. 1 m/s = 1.94384449 knots. Resolution: 1 meter/second.

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
10	Platform Designation	Use EG0104 US Key	Pc	Project ID Code	06 0E 2B 34 01 01 01 01 01 01 20 01 00 00 00 00 (CRC 36601)	Device Designation	String	ISO 646	V	Use Platform Designation String e.g.: 'Predator', 'Reaper', 'Outrider', 'Pioneer', 'IgnatER', 'Warrior', 'Shadow', 'Hunter II', 'Global Hawk', 'Scan Eagle', etc. Value field is Free Text. Maximum 127 characters.
11	Image Source Sensor	Use EG0104 US Key	Sn	Sensor Name	06 0E 2B 34 01 01 01 01 04 20 01 02 01 01 00 00 (CRC 53038)	Image Source Device	String	ISO 646	V	String of image source sensor. E.g.: 'EO Nose', 'EO Zoom (DLTV)', 'EO Spotter', 'IR Mitsubishi PtSi Model 500', 'IR InSb Amber Model TBT', 'LYNX SAR Imagery', 'TESAR Imagery', etc. Value field is Free Text. Maximum 127 characters.
12	Image Coordinate System	Use EG0104 US Key	Ic	Image Coordinate System	06 0E 2B 34 01 01 01 01 07 01 01 01 00 00 00 00 (CRC 32410)	Image Coordinate System	String	ISO 646	V	String of the image coordinate system used. E.g.: 'Geodetic WGS84', 'Geocentric WGS84', 'UTM', 'None', etc.
13	Sensor Latitude	Use EG0104 US Key	Sa	Sensor Latitude	06 0E 2B 34 01 01 01 03 07 01 02 01 02 04 02 00 (CRC 8663)	Device Latitude	Degrees	int32	4	Sensor Latitude. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.
14	Sensor Longitude	Use EG0104 US Key	So	Sensor Longitude	06 0E 2B 34 01 01 01 03 07 01 02 01 02 06 02 00 (CRC 20407)	Device Longitude	Degrees	int32	4	Sensor Longitude. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~84 nano degrees.
15	Sensor True Altitude	Use EG0104 US Key	Sl	Sensor Altitude	06 0E 2B 34 01 01 01 01 07 01 02 01 02 02 00 00 (CRC 13170)	Device Altitude	Meters	uint16	2	Altitude of sensor as measured from Mean Sea Level (MSL). Map 0..(2^16-1) to -900..19000 meters. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.
16	Sensor Horizontal field of View	Use EG0104 US Key	Fv	field of View	06 0E 2B 34 01 01 01 02 04 20 02 01 01 08 00 00 (CRC 23753)	field of View (FOV-Horizontal)	Degrees	uint16	2	Horizontal field of view of selected imaging sensor. Map 0..(2^16-1) to 0..180. Resolution: ~2.7 milli degrees.
17	Sensor Vertical Field of View	06 0E 2B 34 01 01 01 07 04 20 02 01 01 0A 01 00 (CRC 30292)	Vv	Vertical Field of View	x	x	Degrees	uint16	2	Vertical field of view of selected imaging sensor. Map 0..(2^16-1) to 0..180. Resolution: ~2.7 milli degrees. Requires data conversion between LS value and SMPTE Mapped US Key.
18	Sensor Relative Azimuth Angle	06 0E 2B 34 01 01 01 01 0E 01 01 02 04 00 00 00 (CRC 944)	Az	Sensor Relative Azimuth Angle	x	x	Degrees	uint32	4	Relative rotation angle of sensor to platform longitudinal axis. Rotation angle between platform longitudinal axis and camera pointing direction as seen from above the platform. Map 0..(2^32-1) to 0..360. Resolution: ~84 nano degrees.
19	Sensor Relative Elevation Angle	06 0E 2B 34 01 01 01 01	De	Sensor Relative	x	x	Degrees	int32	4	Relative Elevation Angle of sensor to platform longitudinal-transverse plane. Negative angles down.

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
		0E 01 01 02 05 00 00 00 (CRC 29956)		Elevation Angle						Map -(2^31-1)..(2^31-1) to +/-180. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Res: ~84 ndeg.
20	Sensor Relative Roll Angle	06 0E 2B 34 01 01 01 01 0E 01 01 02 06 00 00 00 (CRC 61144)	Ro	Sensor Relative Roll Angle	x	x	Degrees	uint32	4	Relative roll angle of sensor to aircraft platform. Twisting angle of camera about lens axis. Top of image is zero degrees. Positive angles are clockwise when looking from behind camera. Map 0..(2^32-1) to 0..360. Resolution: ~84 nano degrees.
21	Slant Range	Use EG0104 US Key	Sr	Slant Range	06 0E 2B 34 01 01 01 01 07 01 08 01 01 00 00 00 (CRC 16588)	Slant Range	Meters	uint32	4	Slant range in meters. Distance to target. Map 0..(2^32-1) to 0..5000000 meters. 1 nautical mile (knot) = 1852 meters. Resolution: ~1.2 milli meters.
22	Target Width	Use EG0104 US Key	Tw	Target Width	06 0E 2B 34 01 01 01 01 07 01 09 02 01 00 00 00 (CRC 60350)	Target Width	Meters	uint16	2	Target Width within sensor field of view. Map 0..(2^16-1) to 0..10000 meters. 1 meter = 3.2808399 feet. Resolution: ~.16 meters.
23	Frame Center Latitude	Use EG0104 US Key	Ta	Target Latitude	06 0E 2B 34 01 01 01 01 07 01 02 01 03 02 00 00 (CRC 17862)	Frame Center Latitude	Degrees	int32	4	Terrain Latitude of frame center. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.
24	Frame Center Longitude	Use EG0104 US Key	To	Target Longitude	06 0E 2B 34 01 01 01 01 07 01 02 01 03 04 00 00 (CRC 63334)	Frame Center Longitude	Degrees	int32	4	Terrain Longitude of frame center. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~84 nano degrees.
25	Frame Center Elevation	06 0E 2B 34 01 01 01 0A 07 01 02 01 03 16 00 00 (CRC 57054)	Te	Frame Center Elevation	x	x	Meters	uint16	2	Terrain elevation at frame center relative to Mean Sea Level (MSL). Map 0..(2^16-1) to -900..19000 meters. Resolution: ~0.3 meters.
26	Offset Corner Latitude Point 1	Use EG0104 US Key	Rg	SAR Latitude 4	06 0E 2B 34 01 01 01 03 07 01 02 01 03 07 01 00 (CRC 23392)	Corner Latitude Point 1 (Decimal Degrees)	Degrees	int16	2	Frame Latitude, offset for upper left corner. Based on WGS84 ellipsoid. Use with Frame Center Latitude. Map -(2^15-1)..(2^15-1) to +/-0.075. Use -(2^15) as an "error" indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.
27	Offset Corner Longitude Point 1	Use EG0104 US Key	Rh	SAR Longitude 4	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0B 01 00 (CRC 11777)	Corner Longitude Point 1 (Decimal Degrees)	Degrees	int16	2	Frame Longitude, offset for upper left corner. Based on WGS84 ellipsoid. Use with Frame Center Longitude. Map -(2^15-1)..(2^15-1) to +/-0.075. Use -(2^15) as an "error" indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.
28	Offset Corner Latitude Point 2	Use EG0104 US Key	Ra	SAR Latitude 1	06 0E 2B 34 01 01 01 03 07 01 02 01 03 08 01 00	Corner Latitude Point 2 (Decimal Degrees)	Degrees	int16	2	Frame Latitude, offset for upper right corner. Based on WGS84 ellipsoid. Use with Frame Center Latitude. Map -(2^15-1)..(2^15-1) to +/-0.075.

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
					(CRC 30545)					Use -(2^15) as an "error" indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.
29	Offset Corner Longitude Point 2	Use EG0104 US Key	Rb	SAR Longitude 1	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0C 01 00 (CRC 43921)	Corner Longitude Point 2 (Decimal Degrees)	Degrees	int16	2	Frame Longitude, offset for upper right corner. Based on WGS84 ellipsoid. Use with Frame Center Longitude. Map -(2^15-1)..(2^15-1) to +/-0.075. Use -(2^15) as an "error" indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.
30	Offset Corner Latitude Point 3	Use EG0104 US Key	Rc	SAR Latitude 2	06 0E 2B 34 01 01 01 03 07 01 02 01 03 09 01 00 (CRC 16481)	Corner Latitude Point 3 (Decimal Degrees)	Degrees	int16	2	Frame Latitude, offset for lower right corner. Based on WGS84 ellipsoid. Use with Frame Center Latitude. Map -(2^15-1)..(2^15-1) to +/-0.075. Use -(2^15) as an "error" indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.
31	Offset Corner Longitude Point 3	Use EG0104 US Key	Rd	SAR Longitude 2	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0D 01 00 (CRC 40097)	Corner Longitude Point 3 (Decimal Degrees)	Degrees	int16	2	Frame Longitude, offset for lower right corner. Based on WGS84 ellipsoid. Use with Frame Center Longitude. Map -(2^15-1)..(2^15-1) to +/-0.075. Use -(2^15) as an "error" indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.
32	Offset Corner Latitude Point 4	Use EG0104 US Key	Re	SAR Latitude 3	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0A 01 00 (CRC 6449)	Corner Latitude Point 4 (Decimal Degrees)	Degrees	int16	2	Frame Latitude, offset for lower left corner. Based on WGS84 ellipsoid. Use with Frame Center Latitude. Map -(2^15-1)..(2^15-1) to +/-0.075. Use -(2^15) as an "error" indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.
33	Offset Corner Longitude Point 4	Use EG0104 US Key	Rf	SAR Longitude 3	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0E 01 00 (CRC 50673)	Corner Longitude Point 4 (Decimal Degrees)	Degrees	int16	2	Frame Longitude, offset for lower left corner. Based on WGS84 ellipsoid. Use with Frame Center Longitude. Map -(2^15-1)..(2^15-1) to +/-0.075. Use -(2^15) as an "error" indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.
34	Icing Detected	06 0E 2B 34 01 01 01 01 0E 01 01 01 0C 00 00 00 (CRC 26785)	Id	Icing Detected	x	x	Icing Code	uint8	1	Flag for icing detected at aircraft location. 0: Detector off 1: No icing Detected 2: Icing Detected
35	Wind Direction	06 0E 2B 34 01 01 01 01 0E 01 01 01 0D 00 00 00 (CRC 7701)	Wd	Wind Direction	x	x	Degrees	uint16	2	Wind direction at aircraft location. This is the direction the wind is coming from relative to true north. Map 0..(2^16-1) to 0..360. Resolution: ~5.5 milli degrees.
36	Wind Speed	06 0E 2B 34 01 01 01 01 0E 01 01 01 0E 00 00 00	Ws	Wind Speed	x	x	Meters/Second	uint8	1	Wind speed at aircraft location. Map 0..255 to 0..100 meters/second. 1 m/s = 1.94384449 knots. Resolution: ~0.4 meters / second.

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
		(CRC 34249)								
37	Static Pressure	06 0E 2B 34 01 01 01 01 0E 01 01 01 0F 00 00 00 (CRC 62333)	Ps	Static Pressure	x	x	Millibar	uint16	2	Static pressure at aircraft location. Map 0..(2^16-1) to 0..5000 mbar. 1 mbar = 0.0145037738 PSI. Resolution: ~0.08 Millibar
38	Density Altitude	06 0E 2B 34 01 01 01 01 0E 01 01 01 10 00 00 00 (CRC 15412)	Da	Density Altitude	x	x	Meters	uint16	2	Density altitude at aircraft location. Relative aircraft performance metric based on outside air temperature, static pressure, and humidity. Map 0..(2^16-1) to -900..19000 meters. Offset = -900. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.
39	Outside Air Temperature	06 0E 2B 34 01 01 01 01 0E 01 01 01 11 00 00 00 (CRC 19072)	At	Air Temperature	x	x	Celcius	int8	1	Temperature outside of aircraft. -128..127 Degrees Celsius. Resolution: 1 degree celsius.
40	Target Location Latitude	06 0E 2B 34 01 01 01 01 0E 01 01 03 02 00 00 00 (CRC 36472)	x	x	x	x	Degrees	int32	4	Calculated Target latitude. This is the crosshair location if different from frame center. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.
41	Target Location Longitude	06 0E 2B 34 01 01 01 01 0E 01 01 03 03 00 00 00 (CRC 63692)	x	x	x	x	Degrees	int32	4	Calculated Target longitude. This is the crosshair location if different from frame center. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~84 nano degrees.
42	Target Location Elevation	06 0E 2B 34 01 01 01 01 0E 01 01 03 04 00 00 00 (CRC 43489)	x	x	x	x	Meters	uint16	2	Calculated target elevation. This is the crosshair location if different from frame center. Map 0..(2^16-1) to -900..19000 meters. Offset = -900. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.
43	Target Track Gate Width	06 0E 2B 34 01 01 01 01 0E 01 01 03 05 00 00 00 (CRC 57173)	x	x	x	x	Pixels	uint8	1	Tracking gate width (x value) of tracked target within field of view. Closely tied to source video resolution in pixels.
44	Target Track Gate Height	06 0E 2B 34 01 01 01 01 0E 01 01 03 06 00 00 00 (CRC 17545)	x	x	x	x	Pixels	uint8	1	Tracking gate height (y value) of tracked target within field of view. Closely tied to source video resolution in pixels.
45	Target Error Estimate - CE90	06 0E 2B 34 01 01 01 01 0E 01 01 03 07 00 00 00	x	x	x	x	Meters	uint16	2	Circular Error 90 (CE90) is the estimated error distance in the horizontal direction.

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
		(CRC 12861)								Specifies the radius of 90% probability on a plane tangent to the earth's surface. Res: ~0.0624 meters
46	Target Error Estimate – LE90	06 0E 2B 34 01 01 01 01 0E 01 01 03 08 00 00 00 (CRC 59091)	x	x	x	x	Meters	uint16	2	Lateral Error 90 (LE90) is the estimated error distance in the vertical (or lateral) direction. Specifies the interval of 90% probability in the local vertical direction. Res: 0.0625 meters
47	Generic Flag Data 01	06 0E 2B 34 01 01 01 01 0E 01 01 03 01 00 00 00 (CRC 5540)	x	x	x	x	None	uint8	1	Generic Flagged Metadata Position Format msb8..1lsb 1- Laser Range 1on,0off 2- Auto-Track 1on,0off 3- IR Polarity 1blk,0wht 4- Icing detected 1ice,0(off/no ice) 5- Slant Range 1measured, 0calc 6- Image Invalid 1invalid, 0valid 7,8- Use 0
48	Security Local Metadata Set	Use ST0102 US key for Local Sets.	x	x	06 0E 2B 34 02 03 01 01 0E 01 03 03 02 00 00 00 (CRC 40980)	Security Local Metadata Set	None	Set	x	Local set tag to include the ST0102 Local Set Security Metadata items within ST0601. Use the ST0102 Local Set Tags within the ST0601 Tag 0d48. The length field is the size of all ST0102 metadata items to be packaged within Tag 0d48.
49	Differential Pressure	06 0E 2B 34 01 01 01 01 0E 01 01 01 01 00 00 00 (CRC 20775)	x	x	x	x	Millibar	uint16	2	Differential pressure at aircraft location. Measured as the Stagnation/impact/total pressure minus static pressure. Map 0..(2^16-1) to 0..5000 mbar. 1 mbar = 0.0145037738 PSI. Res: ~0.08 mbar
50	Platform Angle of Attack	06 0E 2B 34 01 01 01 01 0E 01 01 01 02 00 00 00 (CRC 51963)	x	x	x	x	Degrees	int16	2	Platform Attack Angle. Angle between platform longitudinal axis and relative wind. Positive angles for upward relative wind. Map -(2^15-1)..(2^15-1) to +/-20. Use -(2^15) as an "out of range" indicator. -(2^15) = 0x8000. Res: ~610 micro degrees.
51	Platform Vertical Speed	06 0E 2B 34 01 01 01 01 0E 01 01 01 03 00 00 00 (CRC 48207)	x	x	x	x	Meters/Second	int16	2	Vertical speed of the aircraft relative to zenith. Positive ascending, negative descending. Map-(2^15-1)..(2^15-1) to +/-180 Use -(2^15) as an "out of range" indicator. -(2^15) = 0x8000. Resolution: ~ 0.0055 meters/second.
52	Platform Sideslip Angle	06 0E 2B 34 01 01 01 01 0E 01 01 01 04 00 00 00 (CRC 60770)	x	x	x	x	Degrees	int16	2	The sideslip angle is the angle between the platform longitudinal axis and relative wind. Positive angles to right wing, neg to left. Map -(2^15-1)..(2^15-1) to +/-20. Use -(2^15) as an "out of range" indicator. -(2^15) = 0x8000.

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
										Res: ~610 micro deg.
53	Airfield Barometric Pressure	06 0E 2B 34 01 01 01 01 0E 01 01 02 02 00 00 00 (CRC 9257)	x	x	x	x	Millibar	uint16	2	Local pressure at airfield of known height. Pilot's responsibility to update. Map 0..(2^16-1) to 0..5000 mbar. 1013.25mbar = 29.92inHg Min/max recorded values of 870/1086mbar. Resolution: ~0.08 Millibar
54	Airfield Elevation	06 0E 2B 34 01 01 01 01 0E 01 01 02 03 00 00 00 (CRC 21149)	x	x	x	x	Meters	uint16	2	Elevation of Airfield corresponding to Airfield Barometric Pressure. Map 0..(2^16-1) to -900..19000 meters. Offset = -900. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.
55	Relative Humidity	06 0E 2B 34 01 01 01 01 0E 01 01 01 09 00 00 00 (CRC 54500)	x	x	x	x	Percent	uint8	1	Relative Humidity at aircraft location. Map 0..(2^8-1) to 0..100. Resolution: ~0.4%.
56	Platform Ground Speed	06 0E 2B 34 01 01 01 01 0E 01 01 01 05 00 00 00 (CRC 39894)	Gv	Platform Ground Speed	x	x	Meters/Second	uint8	1	Speed projected to the ground of an airborne platform passing overhead. 0..255 meters/sec. 1 m/s = 1.94384449 knots. Resolution: 1 meter/second.
57	Ground Range	06 0E 2B 34 01 01 01 01 0E 01 01 01 06 00 00 00 (CRC 10)	Gr	Ground Range	x	x	Meters	uint32	4	Horizontal distance from ground position of aircraft relative to nadir, and target of interest. Dependent upon Slant Range and Depression Angle. Map 0..(2^32-1) to 0..5000000 meters. 1 nautical mile (knot) = 1852 meters. Resolution: ~1.2 milli meters.
58	Platform Fuel Remaining	06 0E 2B 34 01 01 01 01 0E 01 01 01 07 00 00 00 (CRC 30398)	Fr	Platform Fuel Remaining	x	x	Kilogram	uint16	2	Remaining fuel on airborne platform. Metered as fuel weight remaining. Map 0..(2^16-1) to 0..10000 Kilograms. 1 kilogram = 2.20462262 pounds. Resolution: ~.16 kilograms.
59	Platform Call Sign	06 0E 2B 34 01 01 01 01 0E 01 04 01 01 00 00 00 (CRC 4646)	Cs	Platform Call Sign	x	x	String	ISO 646	V	Call Sign of platform or operating unit. Value field is Free Text.
60	Weapon Load	06 0E 2B 34 01 01 01 01 0E 01 01 01 12 00 00 00 (CRC 53596)	Wl	Weapon Load	x	x	uint16	nibble	2	Current weapons stored on aircraft broken into two bytes: [K][L][V] = [0x41][0x02][[byte1][byte2]] [byteN] = [[nib1][nib2]], nib1= msn byte1-nib1 = Station Number byte1-nib2 = Substation Number byte2-nib1 = Weapon Type byte2-nib2 = Weapon Variant
61	Weapon Fired	06 0E 2B 34 01 01 01 01 0E 01 01 01 13 00 00 00 (CRC 42984)	Wf	Weapon Fired	x	x	uint8	nibble	1	Indication when a particular weapon is released. Correlate with Unix Time stamp. Identical format to Weapon Load byte 2: [byteN] = [[nib1][nib2]] nib1 = Station Number nib2 = Substation Number

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
62	Laser PRF Code	06 0E 2B 34 01 01 01 01 0E 01 02 02 01 00 00 00 (CRC 28949)	Lc	Laser PRF Code	x	x	None	uint16	2	A laser's Pulse Repetition Frequency (PRF) code used to mark a target. The Laser PRF code is a three or four digit number consisting of the values 1..8. Only the values 1111..8888 can be used without 0's or 9's.
63	Sensor Field of View Name	06 0E 2B 34 01 01 01 01 0E 01 02 02 02 00 00 00 (CRC 60105)	Vn	Sensor Field of View Name	x	x	List	uint8	1	Names sensor field of view quantized steps. 00 = Ultranarrow 01 = Narrow 02 = Medium 03 = Wide 04 = Ultrawide 05 = Narrow Medium 06 = 2x Ultranarrow 07 = 4x Ultranarrow
64	Platform Magnetic Heading	06 0E 2B 34 01 01 01 01 0E 01 01 01 08 00 00 00 (CRC 41552)	Mh	Platform Magnetic Heading	x	x	Degrees	uint16	2	Aircraft magnetic heading angle. Relative between longitudinal axis and Magnetic North measured in the horizontal plane. Map 0..(2^16-1) to 0..360. Resolution: ~5.5 milli degrees.
65	UAS LS Version Number	06 0E 2B 34 01 01 01 01 0E 01 02 03 03 00 00 00 (CRC 13868)	Iv	ESD ICD Version	x	x	Number	uint8	1	Version number of the UAS LS document used to generate a source of UAS LS KLV metadata. 0 is pre-release, initial release (0601.0), or test data. 1..255 corresponds to document revisions ST0601.1 thru ST0601.255.
66	Target Location Covariance Matrix	06 0E 2B 34 02 05 01 01 0E 01 03 03 14 00 00 00 (CRC 28126)	x	x	x	x	TBD	TBD	TBD	Covariance Matrix of the error associated with a targeted location. Details TBD.
67	Alternate Platform Latitude	06 0E 2B 34 01 01 01 01 0E 01 01 01 14 00 00 00 (CRC 63173)	x	x	x	x	Degrees	int32	4	Alternate Platform Latitude. Represents latitude of platform connected with UAS. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.
68	Alternate Platform Longitude	06 0E 2B 34 01 01 01 01 0E 01 01 01 15 00 00 00 (CRC 32881)	x	x	x	x	Degrees	int32	4	Alternate Platform Longitude. Represents longitude of platform connected with UAS. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~84 nano degrees.
69	Alternate Platform Altitude	06 0E 2B 34 01 01 01 01 0E 01 01 01 16 00 00 00 (CRC 7085)	x	x	x	x	Meters	uint16	2	Altitude of alternate platform as measured from Mean Sea Level (MSL). Represents altitude of platform connec ted with UAS. Map 0..(2^16-1) to -900..19000 meters. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
70	Alternate Platform Name	06 0E 2B 34 01 01 01 01 0E 01 01 01 17 00 00 00 (CRC 27929)	x	x	x	x	String	ISO 646	V	Name of alternate platform connected to UAS. E.g.: 'Apachce', 'Rover', 'Predator', 'Reaper', 'Outrider', 'Pioneer', 'IgnatER', 'Warrior', 'Shadow', 'Hunter II', 'Global Hawk', 'Scan Eagle', etc. Value field is Free Text. Maximum 127 characters.
71	Alternate Platform Heading	06 0E 2B 34 01 01 01 01 0E 01 01 01 18 00 00 00 (CRC 47607)	x	x	x	x	Degrees	uint16	2	Heading angle of alternate platform connected to UAS. Relative between longitudinal axis and True North measured in the horizontal plane. Map 0..(2^16-1) to 0..360. Resolution: ~5.5 milli degrees.
72	Event Start Time - UTC	Use EG0104 US Key	x	Mission Start Time, Date, and Date of Collection	06 0E 2B 34 01 01 01 01 07 02 01 02 07 01 00 00 (CRC 11991)	Event Start Date Time - UTC	Microseconds	uint64	8	Start time of scene, project, event, mission, editing event, license, publication, etc. Represented as the microseconds elapsed since midnight (00:00:00), January 1, 1970. Derived from the POSIX IEEE 1003.1 standard. Resolution: 1 microsecond.
73	RVT Local Set	Use ST0806 RVT LS 16-byte Key.	x	x	06 0E 2B 34 02 0B 01 01 0E 01 03 01 02 00 00 00 (CRC 17945)	Remote Video Terminal Local Set	None	Set	x	Local set tag to include the ST0806 RVT Local Set metadata items within ST0601. Use the ST0806 Local Set Tags within the ST0601 Tag 0d73. The length field is the size of all RVT LS metadata items to be packaged within Tag 0d73.
74	VMTI Data Set	Use ST0903 VMTI LS 16-byte Key.	x	x	06 0E 2B 34 02 0B 01 01 0E 01 03 03 06 00 00 00 (CRC 51307)	Video Moving Target Indicator Local Set	None	Set	x	Local set tag to include the ST0903 VMTI Local Set metadata items within ST0601. Use the ST0903 Local Set Tags within the ST0601 Tag 0d74. The length field is the size of all VMTI LS metadata items to be packaged within Tag 0d74.
75	Sensor Ellipsoid Height	06 0E 2B 34 01 01 01 01 0E 01 02 01 82 47 00 00 (CRC 16670)	x	x	x	x	Meters	uint16	2	Sensor Ellipsoid Height as measured from the reference WGS84 Ellipsoid. Map 0..(2^16-1) to -900..19000 meters. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.
76	Alternate Platform Ellipsoid Height	06 0E 2B 34 01 01 01 01 0E 01 02 01 82 48 00 00 (CRC 27951)	x	x	x	x	Meters	uint16	2	Alternate Platform Ellipsoid Height as measured from the reference WGS84 Ellipsoid. Map 0..(2^16-1) to -900..19000 meters. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.
77	Operational Mode	06 0E 2B 34 01 01 01 01 0E 01 01 03 21 00 00 00 (CRC 8938)	x	x	x	x	None	uint8	1	Indicates the mode of operations of the event portrayed in metadata. Enumerated. 0x00 = "Other" 0x01 = "Operational" 0x02 = "Training" 0x03 = "Exercise" 0x04 = "Maintenance"

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
										0x05 = "Test"
78	Frame Center Height Above Ellipsoid	06 0E 2B 34 01 01 01 01 0E 01 02 03 48 00 00 00 (CRC 18095)	x	x	x	x	Meters	uint16	2	Frame Center Ellipsoid Height as measured from the reference WGS84 Ellipsoid. Map 0..(2^16-1) to -900..19000 meters. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.
79	Sensor North Velocity	06 0E 2B 34 01 01 01 01 0E 01 02 02 7E 00 00 00 (CRC 59278)	x	x	x	x	Meters/Sec	int16	2	Northing velocity of the sensor or platform. Positive towards True North Map-(2^15-1)..(2^15-1) to +/-327 Use -(2^15) as an "out of range" indicator. -(2^15) = 0x8000. Resolution: ~ 1 cm/sec.
80	Sensor East Velocity	06 0E 2B 34 01 01 01 01 0E 01 02 02 7F 00 00 00 (CRC 37178)	x	x	x	x	Meters/Sec	int16	2	Easting velocity of the sensor or platform. Positive towards East. Map-(2^15-1)..(2^15-1) to +/-327 Use -(2^15) as an "out of range" indicator. -(2^15) = 0x8000. Resolution: ~ 1 cm/sec.
81	Image Horizon Pixel Pack	06 0E 2B 34 02 05 01 01 0E 01 03 02 08 00 00 00 (CRC 37658)	x	x	x	x	Pack	Pack		<Tag 81><length> < start x0, start y0 // point p0 end x1, end y1 // point p1 start lat, start lon end lat, end lon >
82	Corner Latitude Point 1 (Full)	Use EG0104 US Key	Rg	SAR Latitude 4	06 0E 2B 34 01 01 01 03 07 01 02 01 03 07 01 00 (CRC 23392)	Corner Latitude Point 1 (Decimal Degrees)	Degrees	int32	4	Frame Latitude for upper left corner. Full Range. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.
83	Corner Longitude Point 1 (Full)	Use EG0104 US Key	Rh	SAR Longitude 4	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0B 01 00 (CRC 11777)	Corner Longitude Point 1 (Decimal Degrees)	Degrees	int32	4	Frame Longitude for upper left corner. Full Range. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~84 nano degrees.
84	Corner Latitude Point 2 (Full)	Use EG0104 US Key	Ra	SAR Latitude 1	06 0E 2B 34 01 01 01 03 07 01 02 01 03 08 01 00 (CRC 30545)	Corner Latitude Point 2 (Decimal Degrees)	Degrees	int32	4	Frame Latitude for upper right corner. Full Range. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.
85	Corner Longitude Point 2 (Full)	Use EG0104 US Key	Rb	SAR Longitude 1	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0C 01 00 (CRC 43921)	Corner Longitude Point 2 (Decimal Degrees)	Degrees	int32	4	Frame Longitude for upper right corner. Full Range. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~84 nano degrees.

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
86	Corner Latitude Point 3 (Full)	Use EG0104 US Key	Rc	SAR Latitude 2	06 0E 2B 34 01 01 01 03 07 01 02 01 03 09 01 00 (CRC 16481)	Corner Latitude Point 3 (Decimal Degrees)	Degrees	int32	4	Frame Latitude for lower right corner. Full Range. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.
87	Corner Longitude Point 3 (Full)	Use EG0104 US Key	Rd	SAR Longitude 2	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0D 01 00 (CRC 40097)	Corner Longitude Point 3 (Decimal Degrees)	Degrees	int32	4	Frame Longitude for lower right corner. Full Range. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~84 nano degrees.
88	Corner Latitude Point 4 (Full)	Use EG0104 US Key	Re	SAR Latitude 3	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0A 01 00 (CRC 6449)	Corner Latitude Point 4 (Decimal Degrees)	Degrees	int32	4	Frame Latitude for lower left corner. Full Range. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.
89	Corner Longitude Point 4 (Full)	Use EG0104 US Key	Rf	SAR Longitude 3	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0E 01 00 (CRC 50673)	Corner Longitude Point 4 (Decimal Degrees)	Degrees	int32	4	Frame Longitude for lower left corner. Full Range. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~84 nano degrees.
90	Platform Pitch Angle (Full)	Use EG0104 US Key	Ip	UAV Pitch (INS)	06 0E 2B 34 01 01 01 07 07 01 10 01 05 00 00 00 (CRC 51059)	Platform Pitch Angle	Degrees	int32	4	Aircraft pitch angle. Angle between longitudinal axis and horizontal plane. Positive angles above horizontal plane. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "out of range" indicator. -(2^31) = 0x80000000. Res: ~42 nano deg.
91	Platform Roll Angle (Full)	Use EG0104 US Key	Ir	UAV Roll (INS)	06 0E 2B 34 01 01 01 07 07 01 10 01 04 00 00 00 (CRC 45511)	Platform Roll Angle	Degrees	int32	4	Platform roll angle. Angle between transverse axis and transvers-longitudinal plane. Positive angles for lowered right wing. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "error" indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.
92	Platform Angle of Attack (Full)	06 0E 2B 34 01 01 01 01 0E 01 01 01 02 00 00 00 (CRC 51963)	x	x	x	x	Degrees	int32	4	Platform Attack Angle. Angle between platform longitudinal axis and relative wind. Positive angles for upward relative wind. Map -(2^31-1)..(2^31-1) to +/-90. Use -(2^31) as an "out of range" indicator. -(2^31) = 0x80000000. Res: ~42 nano deg.
93	Platform Sideslip Angle (Full)	06 0E 2B 34 01 01 01 01 0E 01 01 01 04 00 00 00 (CRC 60770)	x	x	x	x	Degrees	int32	4	Angle between the platform longitudinal axis and relative wind. Full Range. Positive angles to right wing, neg to left. Map -(2^31-1)..(2^31-1) to +/-90.

TAG	LS Name	Mapped US	ESD	ESD Name	US	US Name	Units	Format	Len	Notes
										Use -(2^31) as an "out of range" indicator. -(2^31) = 0x80000000. Res: ~42 nano deg.
94	MIIS Core Identifier	Use ST1204 MIIS Core 16-byte Key.	x	x	06 0E 2B 34 01 01 01 01 0E 01 04 05 03 00 00 00 (CRC 30280)	Motion Imagery Identification System Core	None	Binary Value	x	Local set tag to include the ST1204 MIIS Core Identifier Binary Value within ST0601. Use according to the rules and requirements defined in ST1204.
95	SAR Motion Imagery Metadata	Use ST1206 SARMI 16-byte Key.	x	x	06 0E 2B 34 02 0B 01 01 0E 01 03 03 0D 00 00 00 (CRC 54900)	SAR Motion Imagery Metadata	None	Set	x	Local set tag to include the ST1206 SAR Motion Imagery Metadata Local Set data within ST0601. Use according to the rules and requirements defined in ST1206.

7.2 Platform and Sensor Position and Rotation Metadata

To better assist the understanding and interoperability of the UAS LS, this section describes the collective relationship between the multiple platform and sensor position and rotation metadata items available within the UAS LS.

Together the platform location and attitude, along with the sensor relative pointing angles define the location of an image or image sequence. Metadata items for sensor location (Tags 13, 14, & 15/75), platform rotations (Tags 5, 6, & 7), and sensor rotations (Tags 18, 19, & 20), along with Euler Angle order of operation rules are discussed in more detail in the subsections that follow.

7.2.1 Sensor Location

The metadata items associated with sensor location are:

1. *Latitude* - Sensor Latitude (Tag 13)
2. *Longitude* - Sensor Longitude (Tag 14)
3. Height - Sensor Altitude (Tag 15), or Sensor Ellipsoid Height (Tag 75)

7.2.2 Platform Rotations

The metadata items associated with platform attitude and rotations are:

1. Platform Yaw - Platform Heading Angle (Tag 5)

   The platform heading angle is defined as the angle between the platform longitudinal axis (line made by the fuselage) and true north measured in the horizontal plane. Angles increase in a clockwise direction when looking from above the platform. North is 0 degrees, east is 90, south is 180, and west is 270 degrees from true north.

2. Platform Pitch - Platform Pitch Angle (Tag 6), or full-range Platform Pitch (Tag 90)

   The pitch angle of the platform is the angle between the longitudinal axis (line made by the fuselage) and the horizontal plane. Angles are positive when the platform nose is above the horizontal plane. Take special care for Platform Pitch angles equal to +/- 90.

3. Platform Roll - Platform Roll Angle (Tag 7), or full-range Platform Roll (Tag 91)

   The rotation operation performed about the longitudinal axis forms the roll angle between the previous aircraft transverse-longitudinal plane and the new transverse axis location (line from wing tip to wing tip). Positive angles correspond to the starboard (right) wing lowered below the previous aircraft transverse-longitudinal plane.

7.2.3 Sensor Rotations

The metadata items associated with sensor rotations are:

1. Sensor Relative Yaw - Sensor Relative Azimuth Angle (Tag 18)

   The sensor relative azimuth angle is defined as the angle between the platform longitudinal axis (line made by the fuselage) and the sensor pointing direction, measured in the plane formed by the platform longitudinal and transverse axes (line from wing tip to wing tip).

Angles increase in a clockwise direction when looking from above the platform, with 0 degrees forward along the longitudinal axis.

2. Sensor Relative Pitch - Sensor Relative Elevation Angle (Tag 19)

The relative elevation angle of the sensor to the aircraft is the downward (or upward) pointing angle of the sensor relative to the plane formed by the longitudinal axis (line made by the fuselage) and the transverse axis (line from wing tip to wing tip). Sensor pointing angles below the platform longitudinal-transverse plane are negative.

3. Sensor Relative Roll - Sensor Relative Roll Angle (Tag 20)

Sensors that are able to rotate their camera about the lens axis make use of this sensor relative roll angle. A roll angle of zero degrees occurs when the top and bottom edges of the captured image lie perpendicular to the plane created by the sensor relative depression angle axis. Positive angles are clockwise when looking from behind the camera.

7.2.4 Euler Angle Order of Operations

In order to properly determine the orientation of a sensor on an airborne platform using the UAS LS metadata items outlined in Section 7.2, a specific order of position, and rotation angles must be followed. The order of operations required to determine a sensor's orientation is as follows:

1. Move a sensor to the geodetic Latitude, Longitude, and altitude using
    a. Tag 13, Sensor Latitude
    b. Tag 14, Sensor Longitude
    c. Tag 15, Sensor Altitude (or Tag 75: Sensor Ellipsoid Height)
2. Convert the geodetic coordinates to a geocentric system, then use a local-level North-East-Down (NED, right hand rule) sensor orientation.
3. Perform a Platform Rotation. Start with Yaw, then Pitch, the Roll.
    a. Tag 5: Platform Heading Angle
    b. Tag 6: Platform Pitch Angle
    c. Tag 7: Platform Roll Angle

Refer to Figure 7-1 for the different platform rotations outlined in steps 2 and 3 above.

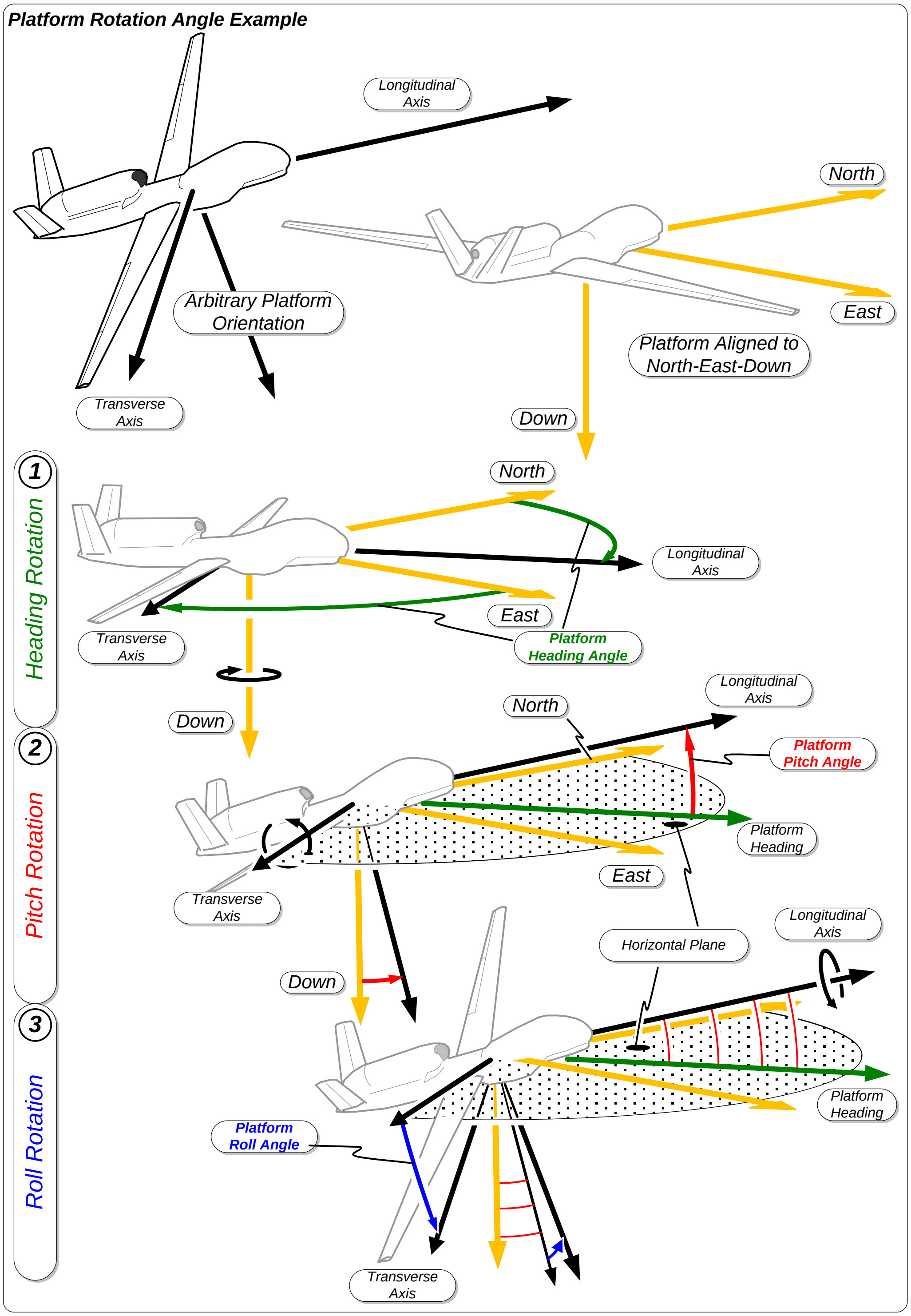

Figure 7-1 : Platform Rotation Angle Example

4. Perform a Sensor Rotation. Start with Yaw, then Pitch, then Roll.
   a. Tag 18: Sensor Relative Azimuth Angle
   b. Tag 19: Sensor Relative Elevation Angle
   c. Tag 20: Sensor Relative Roll Angle

Refer to Figure 7-2 for the different sensor rotations outlined in steps 4 above.

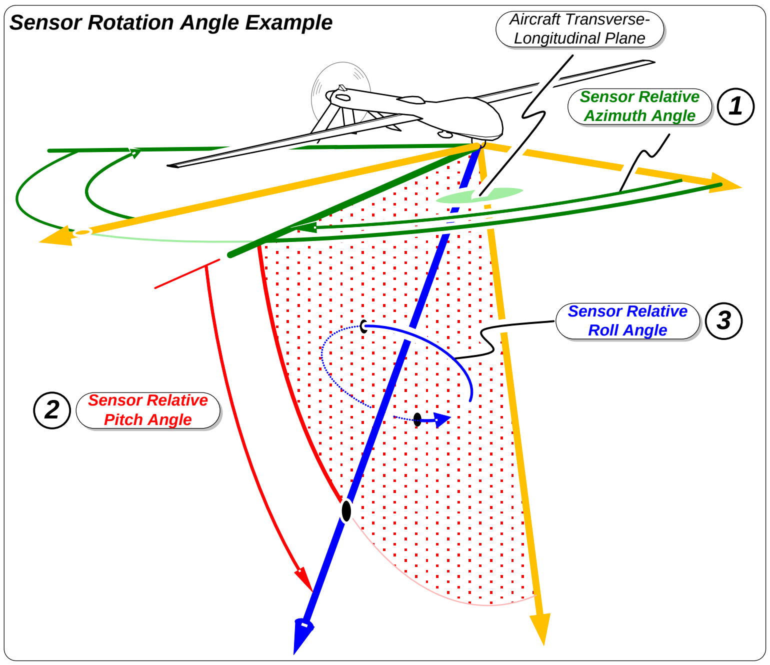

Figure 7-2 : Sensor Rotation Angle Example

Once the platform and sensor attitude is known, the user is free to use other metadata items like horizontal and vertical field of view to suit the purpose of an intended application.

7.3 Sensor Image Geoposition Corner Metadata

An example of corner-coordinate metadata as used in a Motion Imagery system is shown in Figure 7-3 below.

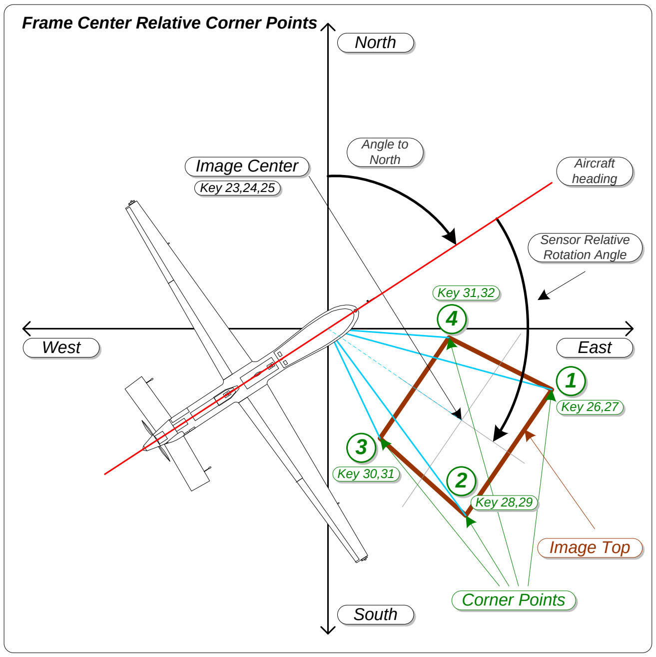

Figure 7-3: Corner Coordinate Metadata

The Sensor Image Corner Latitude/Longitude metadata consists of the items shown in Figure 7-4. Corner coordinates are numbered to conform to National Imagery Transmission Format (NITF) Standard numbering convention for single image frame corner coordinates.

See the NITF Standards document MIL-STD-2500C Version 2.1[17] for more information about corner coordinates. Corners not corresponding to geographic locations, i.e., above the horizon, are not to be included. This numbering scheme is different than the one used in the ESD interface described in ASI-00209 Rev D "Exploitation Support Data (ESD) External Interface Control Document" [14].

Figure 7-4 shows a detailed mapping between metadata items for each corner point.

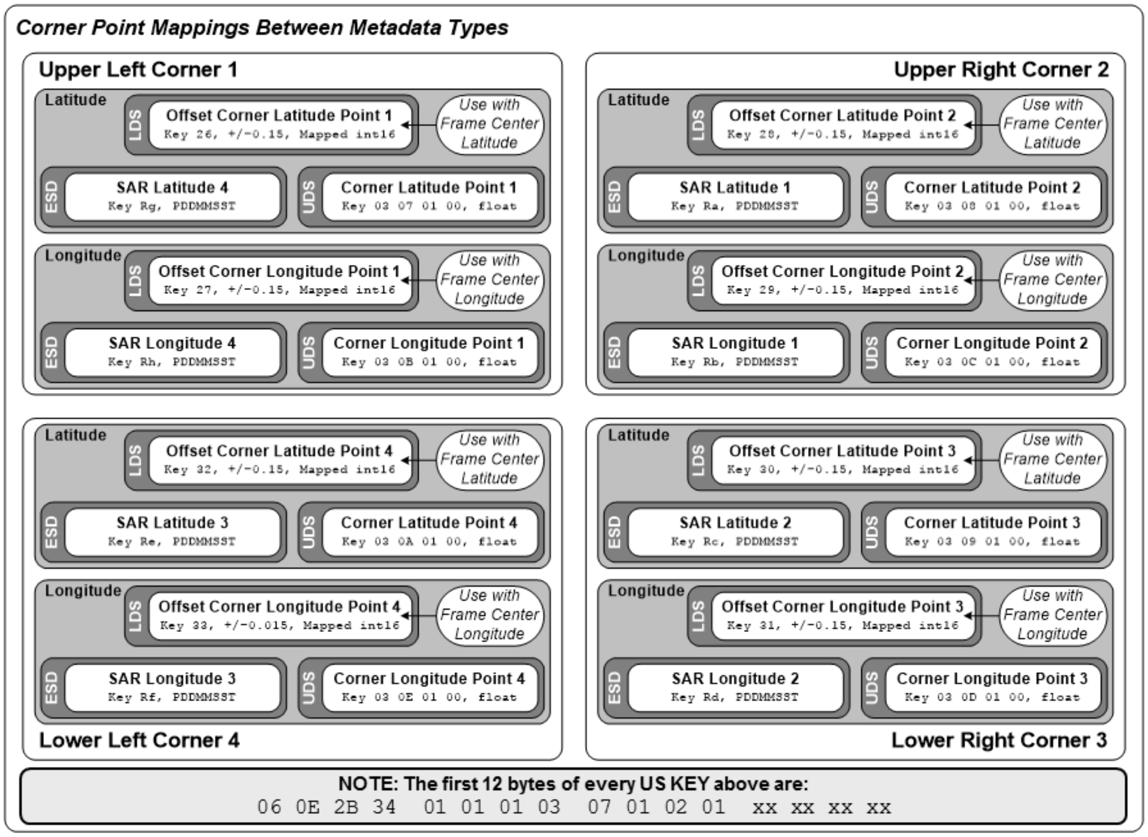

Figure 7-4: Corner Point Mapping

The LS makes use of Offset Corner Point metadata items and requires addition with the LS Frame Center coordinates to determine the actual corner points. This differs from the US and ESD data types which use corner point items that are independent of the frame center items and explicitly define actual corner coordinates without needing computation.

The LS Offset Corner Points use a mapped 2-byte signed integer which is converted to a decimal and added as an offset to the respective decimal representation of LS Frame Center Latitude or Longitude to determine the actual corner point. This offset method used in the LS only covers a finite area about an image center point (16.6km x 16.6km square area at the Equator) yet still adequately represents a typical Motion Imagery sequence while it conserves significant bandwidth over the US implementation. In comparison, each Latitude and Longitude US corner point has one 8-byte floating point value corresponding to decimal degrees which covers the entire globe.

7.4 Alternate Platform Guideline

Within the UAS LS there are multiple metadata items which provide position and other relevant data about an "Alternate Platform". These items differ from the "Platform" or "Sensor" metadata field in that the "Alternate Platform" items provide no position or attitude information about an image sequence to which a UAS LS stream is tied.

Whenever a MISP-compliant Motion Imagery stream is created (a binary sequence typically containing metadata (i.e. UAS LS) and compressed video within an MPEG-2 transport stream) within a sensor/platform system, the sensor and platform metadata field directly relate to the imagery while the "Alternate Platform" field describe an external platform.

For instance, suppose Platform B is receiving a Motion Imagery stream from Platform A. The metadata Platform B receives would describe where Platform A is, as well as its sensor's pointing angles. Should Platform A also include "Alternate Platform" metadata, those metadata field would represent position data for Platform C, or D, or even Platform B, but Platform A must not represent itself within "Alternate Platform" field.

As a general guideline, "Alternate Platform" field do not directly describe a Motion Imagery sequence, but aid situational awareness to a Motion Imagery stream already described through metadata by the host platform.

7.5 Out of Range and Error Values

Various ST 0601 metadata items have special bit-pattern representations which indicate either the item is "Out of Range", or there is an "Error".

For instance, some angles within this Standard (such as platform pitch and roll) are represented as mapped integer values lying between a maximum and minimum angular value. Should the measured angular value lie outside the maximum or minimum value defined in this Standard, the metadata source is given the ability to convey information that a value was measured and is "Out of Range".

Other items such as latitudes and longitudes span entire angular dimensions and are not limited to an artificial minimum by this standard. In this case a single bit sequence is reserved to indicate that the metadata value is an "Error" instead of "Out of Range".

While not all mapped integer metadata items have "Error" or "Out of Range" bit sequences, those that do should only use these special values sparingly.

Systems receiving ST 0601 metadata should also take care when parsing mapped integer items to check for "Error" or "Out of Range" values prior to using the data value being represented.

8 Conversions and Mappings between Metadata Types

Metadata items that are common amongst UAS LS, Predator US, and ESD data formats each convey identical information. However, since each metadata format represents the same metadata items differently (e.g. mapped integer, float, string, etc.), the data resolution between format types is different. This section provides conversions and mappings between LS, US, and ESD metadata items.

Fields marked with an "x" are to be considered not applicable.

Example conversions tables only containing information for the LS do not have equivalent US or ESD representations.

8.1 Tag 1: Checksum Conversion

LS Tag	1	Units	Range	Format
LS Name	Checksum	None	0..(2^16-1)	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 02 03 01 00 00 00 (CRC 56132)			
Notes		**Conversion Formula**		
- Checksum used to detect errors within a UAV Local Set packet. - Lower 16-bits of summation. - Performed on entire LS packet, including 16-byte US key and 1-byte checksum length.		x x		
Example Value		**Example LS Packet**		
0x8C ED		[K][L][V] = [0d1][0d2][0x8C ED]		

8.1.1 Example 16-bit Checksum Code

```
unsigned short bcc_16 (
  unsigned char * buff,  // Pointer to the first byte in the 16-byte UAS LS key.
  unsigned short len )   // Length from 16-byte US key up to 1-byte checksum length.
{
  unsigned short bcc = 0, i;  // Initialize Checksum and counter variables.
  for ( i = 0 ; i < len; i++)
    bcc += buff[i] << (8 * ((i + 1) % 2));
  return bcc;
}  // end of bcc_16 ()
```

8.1.2 Sample Checksum Data

```
64 bits to checksum: 060E 2B34 0200 81BB

    060E
  + 2B34
  ------
    3142
  + 0200
  ------
    3342
  + 81BB
  ------
    B4FD <-- Final Checksum
```

8.2 Tag 2: UNIX Time Stamp Conversion

		Units	Range	Format
LS Tag	2	Microseconds	0..(2^64-1)	uint64
LS Name	UNIX Time Stamp			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Coordinated Universal Time (UTC) represented in the number of microseconds elapsed since midnight (00:00:00), January 1, 1970. - Derived from the POSIX IEEE 1003.1 standard. - Resolution: 1 microsecond.	x x

Example Value	Example LS Packet
Oct. 24, 2008. 00:13:29.913	[K][L][V] = [0d2][0d8][0x00 04 59 F4 A6 AA 4A A8]

US Key	06 0E 2B 34 01 01 01 03 07 02 01 01 01 05 00 00	ESD Digraph	x
US Name	User Defined Time Stamp - microseconds since 1970	ESD Name	x

Units	Range	Format	Units	Range	Format
uSec	uint64	uint64	x	x	x

Notes	Notes
- Time Stamp application defined by user. - 64 bit integer which represents the number of microseconds since Jan 1, 1970 UTC derived from the POSIX (IEEE 1003.1) standard.	- x

US Conversion	ESD Conversion
x	x
To US: - x *To LS:* - x	*To ESD:* - x *To LS:* - x

8.2.1 Example UNIX Time Stamp

This metadata element represents UTC time as the number of microseconds elapsed since the UNIX epoch of January 1, 1970, and is contained within 8-bytes.

A Precision Time Stamp discretely labels a scale of time. This system is widely used within systems of differing underlying architectures. The Precision Time Stamp is an encoding of Coordinated Universal Time (UTC), and therefore, accounts for the addition (or subtraction) of leap seconds. Leap seconds are used to synchronize the UTC clock metric with the yearly rotation period of the earth about the sun.

8.3 *Tag 3: Mission ID Conversion*

LS Tag	3	Units	Range	Format
LS Name	Mission ID	String	1..127	ISO 646
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 04 01 03 00 00 00			
Notes		Conversion Formula		
- Descriptive Mission Identifier to distinguish event or sortie. - Value field is Free Text. - Maximum 127 characters.		x x		
Example Value	Example LS Packet			
MISSION01	[K][L][V] = [0d3][0d9][0x4D 49 53 53 49 4F 4E 30 31]			

US Key	06 0E 2B 34 01 01 01 01 01 05 05 00 00 00 00 00		ESD Digraph	Mn	
US Name	Episode Number		ESD Name	Mission Number	
Units	Range	Format	Units	Range	Format
Number	x	Float	Alpha-Numeric	1..9	String
Notes			Notes		
- x			- Number to distinguish different missions started on a given day.		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			x *To ESD:* - x *To LS:* - x		

8.3.1 Example Mission ID

Format and contents of a Mission ID are to be determined.

8.4 Tag 4: Platform Tail Number Conversion

		Units	Range	Format
LS Tag	4	String	1..127	ISO 646
LS Name	Platform Tail Number			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 04 01 02 00 00 00			

Notes	Conversion Formula
- Identifier of platform as posted. - E.g.: "AF008", "BP101", etc. - Value field is Free Text. - Maximum 127 characters.	x x

Example Value	Example LS Packet
AF-101	[K][L][V] = [0d4][0d6][0x41 46 2D 31 30 31]

US Key	x		ESD Digraph	Pt	
US Name	x		ESD Name	Platform Tail Number	
Units	Range	Format	Units	Range	Format
x	x	x	Number	0..3	N
Notes			Notes		
- x			- x		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			x *To ESD:* - x *To LS:* - x		

8.4.1 Example Platform Tail Number

Format and contents of a Platform Tail Number are to be determined.

8.5 Tag 5: Platform Heading Angle Conversion

LS Tag	5	Units	Range	Format
LS Name	Platform Heading Angle	Degrees	0..360	uint16
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Aircraft heading angle. Relative between longitudinal axis and True North measured in the horizontal plane. - Map 0..(2^16-1) to 0..360. - Resolution: ~5.5 milli degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_5\_dec = \left(\frac{360}{65535} * LS\_5\right)$

Example Value	Example LS Packet
159.9744 Degrees	[K][L][V] = [0d5][0d2][0x71 C2]

US Key	06 0E 2B 34 01 01 01 07 07 01 10 01 06 00 00 00	ESD Digraph	Ih
US Name	Platform Heading Angle	ESD Name	UAV Heading (INS)

Units	Range	Format	Units	Range	Format
Degrees	0..360	Float	Degrees	0..359.99	DDD.HH

Notes	Notes
- Heading angle of platform expressed in degrees. - The Heading of an airborne platform is the angle from True North of its longitudinal axis projected onto the horizontal plane.	- True heading of the aircraft.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{360}{65535} * LS\_uint\right)$ *To US:* - US = (float)(360/0xFFFF * LS) *To LS:* - LS = (uint16)round(0xFFFF/360 * US)	$ESD\_dec = \left(\frac{360}{65535} * LS\_uint\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to uint16.

8.5.1 Example Platform Heading Angle

The platform heading angle is defined as the angle between longitudinal axis (line made by the fuselage) and true north measured in the horizontal plane. Angles increase in a clockwise direction when looking from above the platform. North is 0 degrees, east is 90, south is 180, and west is 270 degrees from true north. Refer to Figure 8-1:

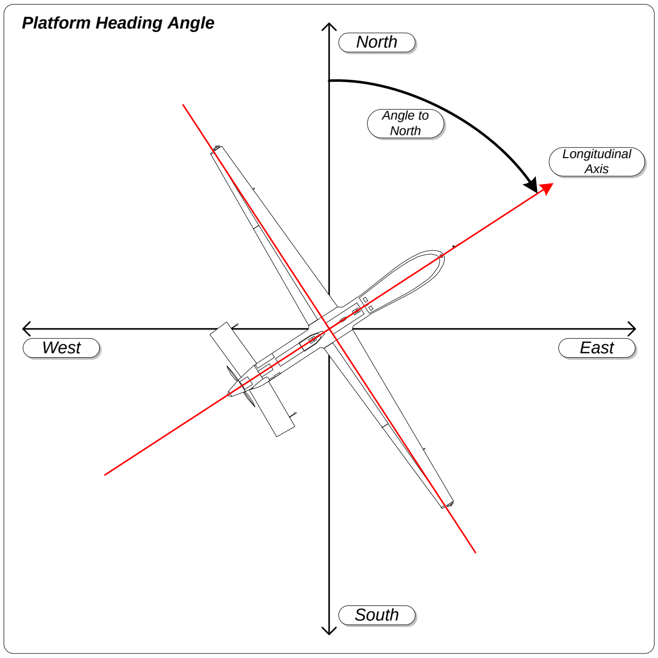

Figure 8-1: Platform True Heading Angle

8.6 Tag 6: Platform Pitch Angle Conversion

LS Tag	6	Units	Range	Format
LS Name	Platform Pitch Angle	Degrees	+/- 20	int16
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Aircraft pitch angle. Angle between longitudinal axis and horizontal plane. - Positive angles above horizontal plane. - Map -(2^15-1)..(2^15-1) to +/-20. - Use -(2^15) as "out of range" indicator. - -(2^15) = 0x8000. - Resolution: ~610 micro degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_06\_dec = \left(\frac{40}{65534} * LS\_int\right)$

Example Value	Example LS Packet
-0.4315251 Degrees	[K][L][V] = [0d6][0d2][0xFD 3D]

US Key	06 0E 2B 34 01 01 01 07 07 01 10 01 05 00 00 00	ESD Digraph	Ip
US Name	Platform Pitch Angle	ESD Name	UAV Pitch (INS)

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Float	Degrees	+/- 20.00	PDD.HH

Notes	Notes
- Pitch angle of platform expressed in degrees. - The Pitch of an airborne platform describes the angle the longitudinal axis makes with the horizontal (i.e., equi-potential gravitational surface);	- Pitch angle of the aircraft.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{40}{65534} * LS\_int\right)$ *To US:* - US = (float)(40/0xFFFE * LS) *To LS:* - LS = (int16)round(0xFFFE/40 * US)	$ESD\_dec = \left(\frac{40}{65534} * LS\_int\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.6.1 Example Platform Pitch Angle

For legacy purposes, both range-restricted (Tag 6) and full-range (Tag 90) representations of Platform Pitch Angle MAY appear in the same ST 0601 packet. A single representation is preferred favoring the full-range version (Tag 90) as per Section 6.3.

The pitch angle of the platform is the angle between the longitudinal axis (line made by the fuselage) and the horizontal plane. Angles are positive when the platform nose is above the horizontal plane (see Figure 8-2).

Pitch angles are limited to +/- 20 degrees to increase metadata resolution within this range. Should the aircraft experience flight maneuvers beyond this range, an “out of range” indication shall be made within this metadata item. Refer to the figure to the right:

Note that the int16 used in the LS value is encoded using two’s complement.

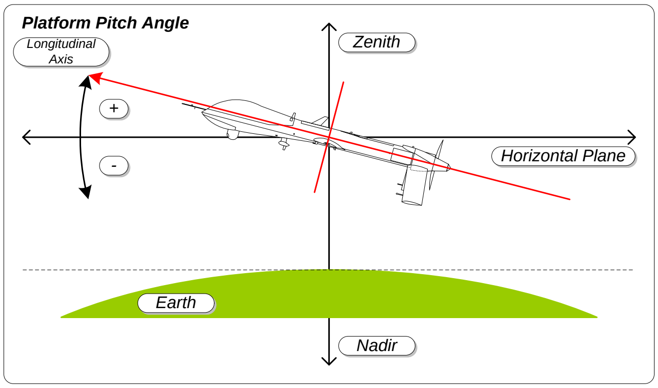

Figure 8-2: Platform Pitch Angle

8.7 Tag 7: Platform Roll Angle Conversion

		Units	Range	Format
LS Tag	7	Degrees	+/- 50	int16
LS Name	Platform Roll Angle			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Platform roll angle. Angle between transverse axis and transvers-longitudinal plane. Positive angles for lowered right wing. - Map (-2^15-1)..(2^15-1) to +/-50. - Use -(2^15) as "out of range" indicator. - -(2^15) = 0x8000. - Res: ~1525 micro deg.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_07\_dec = \left(\frac{100}{65534} * LS\_int\right)$

Example Value	Example LS Packet
3.405814 Degrees	[K][L][V] = [0d7][0d2][0x08 B8]

US Key	06 0E 2B 34 01 01 01 07 07 01 10 01 04 00 00 00	ESD Digraph	Ir
US Name	Platform Roll Angle	ESD Name	UAV Roll (INS)

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Float	Degrees	+/- 50.00	PDD.HH

Notes	Notes
- Roll angle of platform expressed in degrees. - The Roll of an airborne platform is rotation about its longitudinal (front-to-back) axis; - Wings level is zero degrees, positive (negative) angles describe a platform orientation with the right wing down(up).	- Roll angle of the aircraft.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{100}{65534} * LS\_int\right)$	$ESD\_dec = \left(\frac{100}{65534} * LS\_int\right)$
To US: - US = (float)(100/0xFFFE * LS) *To LS:* - LS = (int16)round(0xFFFE/100 * US)	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.7.1 Example Platform Roll Angle

For legacy purposes, both range-restricted (Tag 7) and full-range (Tag 91) representations of Platform Roll Angle MAY appear in the same ST 0601 packet. A single representation is preferred favoring the full-range version (Tag 91) as per Section 6.3.

The rotation operation performed about the longitudinal axis forms the roll angle between the previous aircraft transverse-longitudinal plane and the new transverse axis location (line from wing tip to wing tip). Positive angles correspond to the starboard (right) wing lowered below the previous aircraft transverse-longitudinal plane (see Figure 8-3).

Roll angles are limited to +/- 50 degrees to increase metadata resolution within this range. Should the aircraft experience flight maneuvers beyond this range, an “out of range” indication shall be made within this metadata item. Refer to the figure to the right:

Note that the int16 used in the LS value is encoded using two’s complement.

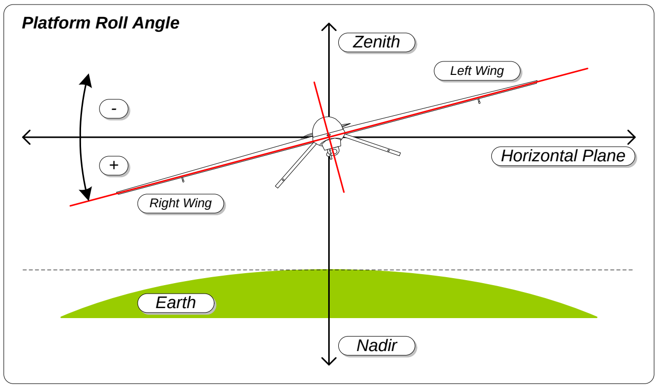

Figure 8-3: Platform Roll Angle

8.8 Tag 8: Platform True Airspeed Conversion

LS Tag	8	Units	Range	Format
LS Name	Platform True Airspeed	Meters/Second	0..255	uint8
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 0A 00 00 00			
Notes		Conversion Formula		
- True airspeed (TAS) of platform. Indicated Airspeed adjusted for temperature and altitude. - 0..255 meters/sec. - 1 m/s = 1.94384449 knots. - Resolution: 1 meter/second.		LS_dec = LS_int LS_08_dec = round(LS_08)		
Example Value		Example LS Packet		
147 m/Sec		[K][L][V] = [0d8][0d1][0x93]		

US Key	x		ESD Digraph	As	
US Name	x		ESD Name	True Airspeed	
Units	Range	Format	Units	Range	Format
x	x	x	Knots	0..999	N
Notes			Notes		
- x			- True airspeed of the aircraft.		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			$ESD\_dec = \left(LS\_uint * \frac{1.94384449\ knots}{1\ meters/second} \right)$ *To ESD:* - Map LS to integer. - Convert integer value to ASCII. *To LS:* - Convert ASCII to integer. - Map integer to uint8.		

8.8.1 Example Platform True Airspeed

True airspeed is the actual speed an aircraft is traveling relative through the air mass in which it flies. Without a relative wind condition, the true airspeed is equal to the speed over the ground. The true airspeed of the aircraft is calculated using the outside temperature, impact pressure (pitot tube), and static pressure.

8.9 Tag 9: Platform Indicated Airspeed Conversion

		Units	Range	Format
LS Tag	9	Meters/Second	0..255	uint8
LS Name	Platform Indicated Airspeed			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 0B 00 00 00			

Notes	Conversion Formula
- Indicated airspeed (IAS) of platform. Derived from Pitot tube and static pressure sensors. - 0..255 meters/sec. - 1 m/s = 1.94384449 knots. - Resolution: 1 meter/second.	LS_dec = LS_int LS_09_dec = round(LS_09)

Example Value	Example LS Packet
159 m/Sec	[K][L][V] = [0d9][0d1][0x9F]

US Key	x	ESD Digraph	Ai
US Name	x	ESD Name	Indicated Airspeed

Units	Range	Format	Units	Range	Format
x	x	x	Knots	0..999	N

Notes	Notes
- x	- Indicated airspeed of the aircraft.

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	$ESD\_dec = \left(LS\_uint * \frac{1.94384449\ knots}{1\ meters/second} \right)$ *To ESD:* - Map LS to integer. - Convert integer value to ASCII. *To LS:* - Convert ASCII to integer. - Map integer to uint8.

8.9.1 Example Platform Indicated Airspeed

The indicated airspeed of an aircraft is calculated from the difference between static pressure, and impact pressure. Static pressure is measured by a sensor not directly in the air stream and impact pressure is measured by a Pitot tube positioned strategically within the air stream. The difference in pressure while moving provides a way to calculate the indicated platform airspeed.

8.10 Tag 10: Platform Designation Conversion

LS Tag LS Name US Mapped Key	10 Platform Designation Use EG0104 US Key	Units: String	Range: 1..127	Format: ISO 646
Notes	- Use Platform Designation String - e.g.: 'Predator', 'Reaper', 'Outrider', 'Pioneer', 'IgnatER', 'Warrior', 'Shadow', 'Hunter II', 'Global Hawk', 'Scan Eagle', etc. - Value field is Free Text. - Maximum 127 characters.	Conversion Formula	x x	
Example Value	MQ1-B	Example LS Packet	[K][L][V] = [0d10][0d5][0x4D 51 31 2D 42]	

US Key US Name	06 0E 2B 34 01 01 01 01 01 01 20 01 00 00 00 00 Device Designation	ESD Digraph ESD Name	Pc Project ID Code	
Units: String	Range: 1..32	Format: ISO 646	Units: Number	Range: 0..99	Format: N
Notes	- Identifies the "house name" of the device used in capturing or generating the essence. - 32 characters maximum. - ISO7 character set.	Notes	- The Project ID of the Collection Platform. - (e.g., Predator, Outrider, Pioneer, etc.)	
US Conversion	x *To US:* - x *To LS:* - x	ESD Conversion	x *To ESD:* - Convert string to Project ID Code. *To LS:* - Convert Project ID Code to string.	

8.10.1 Example Platform Designation

The platform designation metadata item distinguishes which platform is carrying the Motion Imagery generating payload equipment. Some current platforms are shown in Figure 8-4:

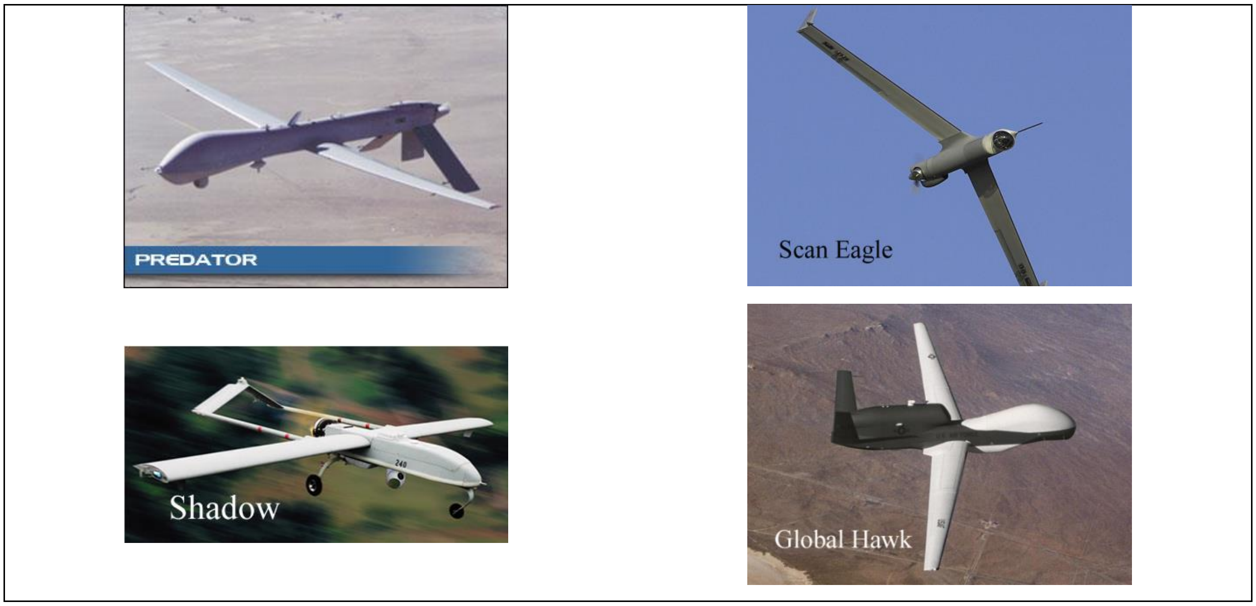

Figure 8-4: Example Platforms

Note: Some systems use the US key `06 0E 2B 34 01 01 01 03 01 01 21 01 00 00 00 00` to represent Platform Designation instead of the 16-byte key shown above (Device Designation) as used in EG 0104.5.

8.11 Tag 11: Image Source Sensor Conversion

		Units	Range	Format
LS Tag	11	String	1..127	ISO 646
LS Name	Image Source Sensor			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- String of image source sensor. - E.g.: 'EO Nose', 'EO Zoom (DLTV)', 'EO Spotter', 'IR Mitsubishi PtSi Model 500', 'IR InSb Amber Model TBT', 'LYNX SAR Imagery', 'TESAR Imagery', etc. - Value field is Free Text. - Maximum 127 characters.	x x

Example Value	Example LS Packet
EO	[K][L][V] = [0d11][0d2][0x45 4F]

US Key	06 0E 2B 34 01 01 01 01 04 20 01 02 01 01 00 00	ESD Digraph	Sn
US Name	Image Source Device	ESD Name	Sensor Name

Units	Range	Format	Units	Range	Format
String	1..32	ISO 646	Name Code	0..7	N

Notes	Notes
- Indicates the type of the image source. - 32 characters maximum. - ISO7 character set.	- Identifies the source of the video image. - 0: EO Nose - 1: EO Zoom (DLTV) - 2: EO Spotter - 3: IR Mitsubishi PtSi Model 500 - 4: IR Mitsubishi PtSi Model 600 - 5: IR InSb Amber Model TBD - 6: Lynx SAR Imagery - 7: TESAR Imagery

US Conversion	ESD Conversion
x	x
To US: - x *To LS:* - x	*To ESD:* - Convert string to ID code. *To LS:* - Convert ID code to string.

8.11.1 Example Image Source Sensor

A sample imaging source sensor is shown in Figure 8-5:

Figure 8-5: Sample Imaging Sensor

8.12 Tag 12: Image Coordinate System Conversion

LS Tag	12	Units	Range	Format
LS Name	Image Coordinate System	String	1..127	ISO 646
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- String of the image coordinate system used. - E.g.: 'Geodetic WGS84', 'Geocentric WGS84', 'UTM', 'None', etc.	x x

Example Value	Example LS Packet
WGS-84	[K][L][V] = [0d12][0d6][0x57 47 53 2D 38 34]

US Key	06 0E 2B 34 01 01 01 01 07 01 01 01 00 00 00 00		ESD Digraph	Ic	
US Name	Image Coordinate System		ESD Name	Image Coordinate System	
Units	Range	Format	Units	Range	Format
String	1..4	ISO 646	Code	0..3	N

Notes	Notes
- Identifies the Digital Geographic Information Exchange Standard (DIGEST) geo-referenced coordinate system used at image capture. - ISO7 character set.	- Identifies the image coordinate system used. - 0: Geodetic WGS84 - 1: Geocentric WGS 84 - 2: UTM - 3: None

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	x *To ESD:* - Convert string to ID code. *To LS:* - Convert ID code to string.

8.12.1 World Geodetic System – 1984 (WGS 84)

The World Geodetic System of 1984 (WGS 84) is a 3-D, Earth-centered reference system developed originally by the U.S. Defense Mapping Agency. This system is the official GPS reference system.

8.12.2 Universal Transverse Mercator (UTM)

UTM is the projection of the earth onto a cylinder. The Universal Transverse Mercator Projection (UTM) divides the globe, excluding the extreme polar areas, into 100km x100km sections and projects each section onto a separate plane that is tangent to the globe at a point within that section. An orthorectifying grid is applied to the projection and results in very minor distortions as no location is greater than 140 km from the point of tangency. Distances, angles and shapes are very accurately depicted within each plane using this earth coordinate system.

Applications exist which convert between UTM and WGS84 coordinate systems and their different datum references.

8.12.3 Notes and Clarification

As of Standard 0601.4, a reference to "DIGEST V2.1 Part 3 Sec 6.4" within the UAS LS section has been removed due to the reference's inapplicability to the Image Coordinate System metadata item.

“Geodetic WGS84” is the preferred Image Coordinate System. “UTM” and other values are provided for sake of completeness to map items between legacy metadata sets.

8.13 Tag 13: Sensor Latitude Conversion

LS Tag	13	Units	Range	Format
LS Name	Sensor Latitude	Degrees	+/- 90	int32
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Sensor Latitude. Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-90. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~42 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_13\_dec = \left(\frac{180}{4294967294} * LS\_13\right)$

Example Value	Example LS Packet
60.1768229669783 Degrees	[K][L][V] = [0d13][0d4][0x55 95 B6 6D]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 02 04 02 00	ESD Digraph	Sa
US Name	Device Latitude	ESD Name	Sensor Latitude

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes (US)	Notes (ESD)
- Specifies a sensor's geographic location in decimal degrees of latitude. - Positive values indicate northern hemisphere. - Negative values indicate southern hemisphere.	- Latitude of the aircraft. + Means North Latitude. All Latitude coordinates use WGS84.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$	$ESD\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$
To US: - US = (double)(180/0xFFFFFFFE * LS) *To LS:* - LS = (int32)round(0xFFFFFFFE/180 * US)	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int32.

8.13.1 Example Latitude

Latitude is the angular distance north or south of the earth's equator, measured in degrees along a meridian. Generated from GPS/INS information and based on the WGS84 coordinate system.

Note that this LS item for Sensor Latitude represents the imaging sensor location versus the aircraft position as represented by the ESD digraph.

In a realized system, this LS item takes into account the lever arm distance between a platform’s GPS antenna (or known central platform position) to a sensor’s general location (like the center of a gimbaled sensor).

While accounting for a lever arm in this crude way is sufficient for many Motion Imagery systems, it is recommended for the user to explore use of Photogrammetric metadata sets (i.e. MISB ST 0801 [21]) for improved representations of system accuracies.

Note that the int32 used in the LS value is encoded using two’s complement.

8.14 Tag 14: Sensor Longitude Conversion

LS Tag	14	Units	Range	Format
LS Name	Sensor Longitude	Degrees	+/- 180	int32
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Sensor Longitude. Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-180. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~84 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_14\_dec = \left(\frac{360}{4294967294} * LS\_14\right)$

Example Value	Example LS Packet
128.426759042045 Degrees	[K][L][V] = [0d14][0d4][0x5B 53 60 C4]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 02 06 02 00	ESD Digraph	So
US Name	Device Longitude	ESD Name	Sensor Longitude

Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes	Notes
- Specifies a sensor's geographic location in decimal degrees of longitude. - Positive values indicate eastern hemisphere. - Negative values indicate western hemisphere.	- Longitude of the aircraft. + Means East Longitude. All Longitude coordinates use WGS84.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$	$ESD\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$
To US: - US = (double)(360/0xFFFFFFFE * LS) *To LS:* - LS = (int32)round(0xFFFFFFFE/360 * US)	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int32.

8.14.1 Example Longitude

Longitude is the angular distance on the earth's surface, measured east or west from the prime meridian at Greenwich, England, to the meridian passing through a position of interest. Generated from GPS/INS information and based on the WGS84 coordinate system.

Note that this LS item for Sensor Longitude represents the imaging sensor location versus the aircraft position as represented by the ESD digraph.

In a realized system, this LS item takes into account the lever arm distance between a platform’s GPS antenna (or known central platform position) to a sensor’s general location (like the center of a gimbaled sensor).

While accounting for a lever arm in this crude way is sufficient for many Motion Imagery systems, it is recommended for the user to explore use of Photogrammetric metadata sets (i.e. MISB ST 0801) for improved representations of system accuracies.

Note that the int32 used in the LS value is encoded using two’s complement.

8.15 Tag 15: Sensor True Altitude Conversion

LS Tag	15	Units	Range	Format
LS Name	Sensor True Altitude	Meters	-900..19000	uint16
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Altitude of sensor as measured from Mean Sea Level (MSL). - Map 0..(2^16-1) to -900..19000 meters. - 1 meter = 3.2808399 feet. - Resolution: ~0.3 meters.	$LS\_dec = \left(\frac{LS\ range}{uint\_range} * LS\_uint\right) - Offset$ $LS\_15\_dec = \left(\frac{19900}{65535} * LS\_15\right) - 900$

Example Value	Example LS Packet
14190.72 Meters	[K][L][V] = [0d15][0d2][0xC2 21]

US Key	06 0E 2B 34 01 01 01 01 07 01 02 01 02 02 00 00		ESD Digraph	Sl	
US Name	Device Altitude		ESD Name	Sensor Altitude	
Units	Range	Format	Units	Range	Format
Meters	Float	Float	Feet	+/- 0..99,999	PN

Notes	Notes
- Altitude of sensor as measured from Mean Sea Level (MSL), (default metres).	- Altitude of the aircraft (MSL).

US Conversion	ESD Conversion
$US\_dec = \left(\frac{19900}{65535} * LS\_uint\right) - 900$	$ESD\_dec = \left(\frac{19900}{65535}*LS\_uint-900\right) * \frac{3.2808399ft}{1m}$
To US: - US = (float)((19900/0xFFFF) * LS - 900) *To LS:* - LS = (uint16)round(0xFFFF/19900 * (US + 900))	*To ESD:* - Convert LS to decimal. - Account for units. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Account for units. - Map decimal to uint16.

8.15.1 Example True Altitude

For legacy purposes, both MSL (Tag 15) and HAE (Tag 75) representations of Sensor True Altitude MAY appear in the same ST 0601 packet. A single representation is preferred favoring the HAE version (Tag 75).

True Altitude is the true vertical distance above mean sea level.

For improved modeling accuracy it is suggested to alternatively use Sensor Ellipsoid Height (Tag 75) should GPS be used to determine altitude.

Note that this LS item for Sensor Altitude represents the imaging sensor location versus the aircraft position as represented by the ESD digraph.

In a realized system, this LS item takes into account the lever arm distance between a platform's GPS antenna (or known central platform position) to a sensor's general location (like the center of a gimbaled sensor).

While accounting for a lever arm in this crude way is sufficient for many Motion Imagery systems, it is recommended for the user to explore use of Photogrammetric metadata sets (i.e. MISB ST 0801) for improved representations of system accuracies.

8.16 Tag 16: Sensor Horizontal field of View Conversion

LS Tag	16	Units	Range	Format
LS Name	Sensor Horizontal field of View	Degrees	0..180	uint16
US Mapped Key	Use EG0104 US Key			

Notes

- Horizontal field of view of selected imaging sensor.
- Map 0..(2^16-1) to 0..180.
- Resolution: ~2.7 milli degrees.

Conversion Formula

$$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$$

$$LS\_16\_dec = \left(\frac{180}{65535} * LS\_16\right)$$

Example Value	Example LS Packet
144.5713 Degrees	[K][L][V] = [0d16][0d2][0xCD 9C]

US Key	06 0E 2B 34 01 01 01 02 04 20 02 01 01 08 00 00	ESD Digraph	Fv
US Name	field of View (FOV-Horizontal)	ESD Name	field of View

Units	Range	Format	Units	Range	Format
Degrees	0..180	Float	Degrees	0..180.00	DDD.HH

Notes (US)

- Sensor Horizontal field of view.

Notes (ESD)

- Angle of view of the lens on the selected camera. Horizontal, across baseline of image, projected onto the terrain (flat terrain model at DTED or other best available elevation data).

US Conversion

$$US\_dec = \left(\frac{180}{65535} * LS\_uint\right)$$

To US:
- US = (float)(180/0xFFFF * LS)

To LS:
- LS = (uint16)round(0xFFFF/180 * US)

ESD Conversion

$$ESD\_dec = \left(\frac{180}{65535} * LS\_uint\right)$$

To ESD:
- Convert LS to decimal.
- Convert decimal to ASCII.

To LS:
- Convert ESD ASCII to decimal.
- Map decimal to uint16.

8.16.1 Example Sensor Horizontal Field of View

The field of view of a lens is defined as the angle over the focal plane where objects are recorded on a film or electro-optical sensor. Field of view is dependent upon the focal length of the lens, and the physical size of the sensor. Typical imaging devices have a square or rectangular imaging sensor. The image (or sequence of images) is typically captured as a square or rectangle and displayed to a user with image edges perpendicular to level sight.

The distance between left edge and right edge is represented as an angle in the horizontal field of view metadata item. Refer to Figure 8-6:

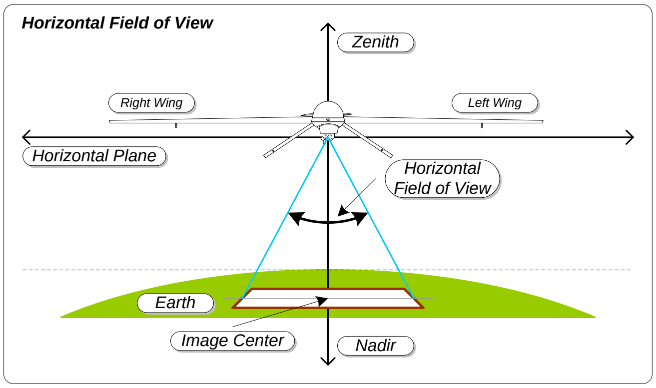

Figure 8-6: Horizontal Field of View

8.17 Tag 17: Sensor Vertical Field of View Conversion

		Units	Range	Format
LS Tag	17	Degrees	0..180	uint16
LS Name	Sensor Vertical Field of View			
US Mapped Key	06 0E 2B 34 01 01 01 07 04 20 02 01 01 0A 01 00			

Notes	Conversion Formula
- Vertical field of view of selected imaging sensor. - Map 0..(2^16-1) to 0..180. - Resolution: ~2.7 milli degrees. - Requires data conversion between LS value and SMPTE Mapped US Key.	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_17\_dec = \left(\frac{180}{65535} * LS\_17\right)$

Example Value	Example LS Packet
152.6436 Degrees	[K][L][V] = [0d17][0d2][0xD9 17]

US Key	x	ESD Digraph	Vv
US Name	x	ESD Name	Vertical Field of View

Units	Range	Format	Units	Range	Format
x	x	x	Degrees	0..180.00	DDD.HH

Notes	Notes
- x	- Angle of view of the lens on the selected camera. Vertical across baseline of image, projected onto the terrain (flat terrain model at DTED or other best available elevation data).

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	$ESD\_dec = \left(\frac{180}{65535} * LS\_uint\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Map decimal to uint16.

8.17.1 Example Sensor Vertical Field of View

The field of view of a lens is defined as the angle over the focal plane where objects are recorded on a film or electro-optical sensor. Field of view is dependent upon the focal length of the lens, and the physical size of the sensor. Typical imaging devices have a square or rectangular imaging sensor. The image (or sequence of images) is typically captured as a square or rectangle and displayed to a user with image edges perpendicular to level sight.

The distance between top edge and bottom edge is represented as an angle in the vertical field of view metadata item. Refer to Figure 8-7:

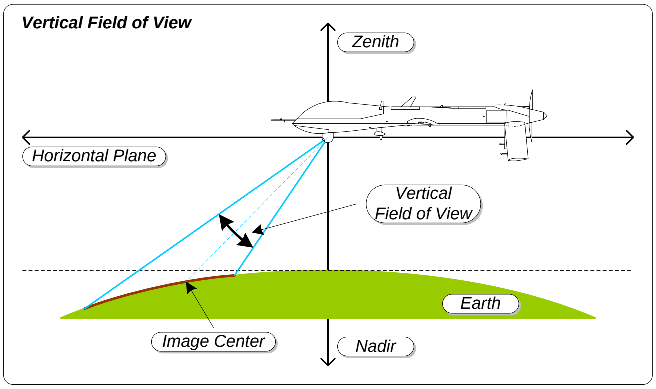

Figure 8-7: Vertical Field of View

8.18 Tag 18: Sensor Relative Azimuth Angle Conversion

		Units	Range	Format
LS Tag	18	Degrees	0..360	uint32
LS Name	Sensor Relative Azimuth Angle			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 02 04 00 00 00			

Notes	Conversion Formula
- Relative rotation angle of sensor to platform longitudinal axis. Rotation angle between platform longitudinal axis and camera pointing direction as seen from above the platform. - Map 0..(2^32-1) to 0..360. - Resolution: ~84 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_18\_dec = \left(\frac{360}{4294967295} * LS\_18\right)$

Example Value	Example LS Packet
160.719211474396 Degrees	[K][L][V] = [0d18][0d4][0x72 4A 0A 20]

US Key	x		ESD Digraph	Az
US Name	x		ESD Name	Sensor Relative Azimuth Angle

Units	Range	Format	Units	Range	Format
x	x	x	Degrees	0..359.99	DDD.HH

Notes	Notes
- x	- Relative rotation angle of sensor to aircraft platform in azimuth. Rotation angle between aircraft fuselage chord and camera pointing direction as seen from above the platform.

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	$ESD\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Map decimal to uint32.

8.18.1 Example Sensor Relative Azimuth Angle

The relative rotation angle of the sensor is the angle formed between the platform longitudinal axis (line made by the fuselage) and the senor pointing direction as measured in the plane formed by the platform longitudinal and transverse axis (line from wing tip to wing tip). Refer to Figure 8-8

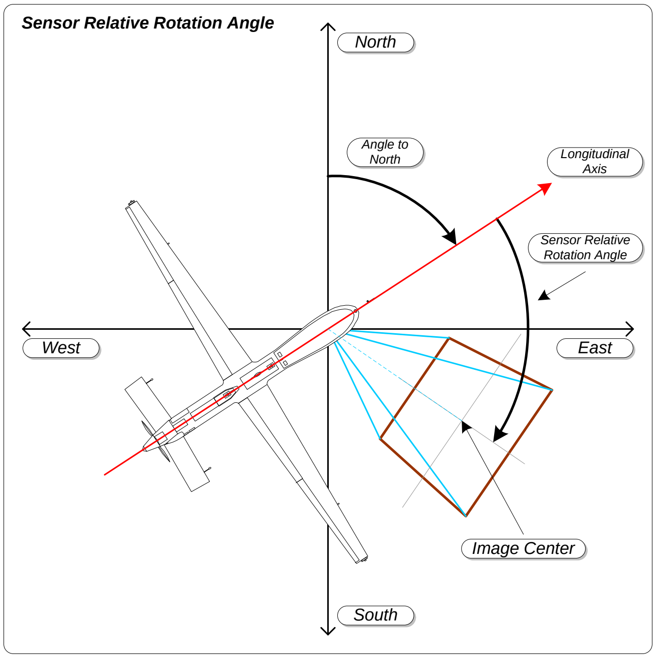

Figure 8-8: Relative Rotation Angle

8.19 Tag 19: Sensor Relative Elevation Angle Conversion

LS Tag	19	Units	Range	Format
LS Name	Sensor Relative Elevation Angle	Degrees	+/- 180	int32
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 02 05 00 00 00			

Notes	Conversion Formula
- Relative Elevation Angle of sensor to platform longitudinal-transverse plane. Negative angles down. - Map -(2^31-1)..(2^31-1) to +/-180. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Res: ~84 ndeg.	$LS\_dec = \left(\frac{LS\ range}{int\_range} * LS\_int\right)$ $LS\_19\_dec = \left(\frac{360}{4294967294} * LS\_19\right)$

Example Value	Example LS Packet
-168.792324833941 Degrees	[K][L][V] = [0d19][0d4][0x87 F8 4B 86]

US Key	x	ESD Digraph	De
US Name	x	ESD Name	Sensor Relative Elevation Angle

Units	Range	Format	Units	Range	Format
x	x	x	Degrees	+/- 180.00	PDDD.HH

Notes	Notes
- x	- Relative Elevation Angle of sensor to aircraft platform. Level flight with camera pointing forward is zero degrees. Negative angles down.

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	$ESD\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Map decimal to uint32.

8.19.1 Example Sensor Relative Elevation Angle

The relative elevation angle of the sensor to the aircraft is the downward (or upward) pointing angle of the sensor relative to the plane formed by the longitudinal axis (line made by the fuselage) and the transverse axis (line from wing tip to wing tip). Sensor pointing angles below the platform longitudinal-transverse plane are negative. Refer to Figure 8-9:

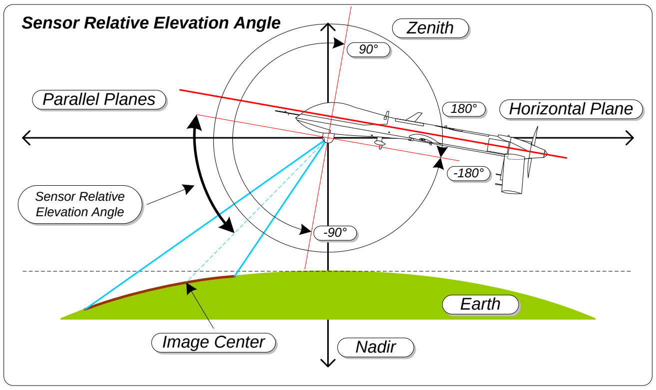

Figure 8-9: Sensor Relative Elevation Angle

Note that the int32 used in the LS value is encoded using two's complement.

8.20 Tag 20: Sensor Relative Roll Angle Conversion

LS Tag	20	Units	Range	Format
LS Name	Sensor Relative Roll Angle	Degrees	0..360	uint32
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 02 06 00 00 00			

Notes	Conversion Formula
- Relative roll angle of sensor to aircraft platform. Twisting angle of camera about lens axis. Top of image is zero degrees. Positive angles are clockwise when looking from behind camera. - Map 0..(2^32-1) to 0..360. - Resolution: ~84 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_20\_dec = \left(\frac{360}{4294967295} * LS\_20\right)$

Example Value	Example LS Packet
176.865437690572 Degrees	[K][L][V] = [0d20][0d4][0x7D C5 5E CE]

US Key	x	ESD Digraph	Ro
US Name	x	ESD Name	Sensor Relative Roll Angle

Units	Range	Format	Units	Range	Format
x	x	x	Degrees	0..359.99	DDD.HH

Notes	Notes
- x	- Relative roll angle of sensor to aircraft platform. Twisting angle of camera about lens axis. Top of image is zero degrees. Positive angles are clockwise when looking from behind camera.

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	$ESD\_dec = \left(\frac{360}{4294967294} * LS\_uint\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Map decimal to uint32.

8.20.1 Example Sensor Relative Roll Angle

Sensors that are able to rotate their camera about the lens axis make use of this sensor relative roll angle. A roll angle of zero degrees occurs when the top and bottom edges of the captured image lie perpendicular to the plane created by the sensor relative depression angle axis. Positive angles are clockwise when looking from behind the camera.

8.21 Tag 21: Slant Range Conversion

		Units	Range	Format
LS Tag	21	Meters	0..5,000,000	uint32
LS Name	Slant Range			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Slant range in meters. Distance to target. - Map 0..(2^32-1) to 0..5000000 meters. - 1 nautical mile (knot) = 1852 meters. - Resolution: ~1.2 milli meters.	$LS\_dec = \left(\frac{LS\ range}{uint\_range} * LS\_uint\right)$ $LS\_dec = \left(\frac{5000000}{4294967295} * LS\_uint\right)$

Example Value	Example LS Packet
68590.98 Meters	[K][L][V] = [0d21][0d4][0x03 83 09 26]

US Key	06 0E 2B 34 01 01 01 01 07 01 08 01 01 00 00 00	ESD Digraph	Sr
US Name	Slant Range	ESD Name	Slant Range

Units	Range	Format	Units	Range	Format
Meters	Float	Float	Nautical Miles	0..18.00	II.HH

Notes	Notes
- Distance from the sensor to the center point on ground of the framed subject (image) depicted in the captured essence, (default metres)	- Distance between the sensor and the target.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{5000000}{4294967295} * LS\_uint\right)$	$ESD\_dec = \left(\frac{5000000}{4294967295} * LS\_uint\right) * \frac{1852knot}{1m}$
To US: - US = (float)(5000000/0xFFFFFFFF * LS) *To LS:* - LS = (uint32)round(0xFFFFFFFF/5000000 * US)	*To ESD:* - Convert LS to decimal. - Account for units. - Convert knots to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Account for units. - Convert feet to uint32.

8.21.1 Example Sensor Slant Range

The slant range is the distance between the sensor and image center. Refer to Figure 8-10.

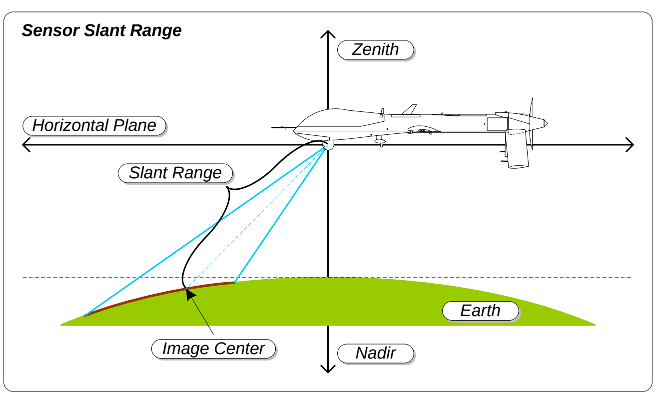

Figure 8-10: Sensor Slant Range

As of ST 0601.3 Generic Flag Data 01 (Tag 47) contains a flag which indicates weather Slant Range is “Computed” or “Measured”. By default the Slant Range is set to “Computed”. “Measured” is to be used when a ranging device (radar, or laser) is providing Slant Range estimates.

8.22 Tag 22: Target Width Conversion

LS Tag	22	Units	Range	Format
LS Name	Target Width	Meters	0..10,000	uint16
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Target Width within sensor field of view. - Map 0..(2^16-1) to 0..10000 meters. - 1 meter = 3.2808399 feet. - Resolution: ~.16 meters.	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_dec = \left(\frac{10000}{65535} * LS\_uint\right)$

Example Value	Example LS Packet
722.8199 Meters	[K][L][V] = [0d22][0d2][0x12 81]

US Key	06 0E 2B 34 01 01 01 01 07 01 09 02 01 00 00 00	ESD Digraph	Tw
US Name	Target Width	ESD Name	Target Width

Units	Range	Format	Units	Range	Format
Meters	Float	Float	Feet	0..99,999	N

Notes	Notes
- Horizontal half width of the target frame image; used to compute the four corner points of the frame, (default metres)	- Width of the EO/IR Payloads field of view on the ground

US Conversion	ESD Conversion
$US\_dec = \left(\frac{10000}{65535} * LS\_uint\right)$	$ESD\_dec = \left(\frac{10000}{65535} * LS\_uint\right) * \frac{3.2808399ft}{1m}$
To US: - US = (float)(10000/0xFFFF * LS) *To LS:* - LS = (uint16)round(0xFFFF/10000 * US)	*To ESD:* - Convert LS to decimal. - Account for units. - Convert feet to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Account for units. - Convert meters to uint32.

8.22.1 Example Sensor Target Width

The target width is the linear ground distance between the center of both sides of the captured image. Refer to Figure 8-11.

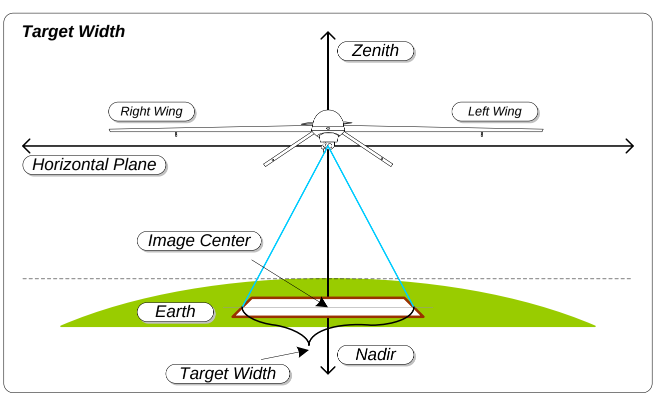

Figure 8-11: Target Width

Note: SMPTE periodically makes updates to its use of metadata keys and has made a change denoting Target Width as the half-width of the image. Despite this change in the SMPTE definition, the MISB continues to interpret Target Width for ST 0601 as full-width.

8.23 Tag 23: Frame Center Latitude Conversion

LS Tag	23	Units	Range	Format
LS Name	Frame Center Latitude	Degrees	+/- 90	int32
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Terrain Latitude of frame center. Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-90. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~42 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_23\_dec = \left(\frac{180}{4294967294} * LS\_23\right)$

Example Value	Example LS Packet
-10.5423886331461 Degrees	[K][L][V] = [0d23][0d4][0xF1 01 A2 29]

US Key	06 0E 2B 34 01 01 01 01 07 01 02 01 03 02 00 00	ESD Digraph	Ta
US Name	Frame Center Latitude	ESD Name	Target Latitude

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes	Notes
- Specifies the video frame center point geographic location in decimal degrees of latitude. - Positive values indicate northern hemisphere. - Negative values indicate southern hemisphere.	- Latitude of the EO/IR payload's aimpoint on the ground. + Means North lattitude. All latitude coordinates use WGS84.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$	$ESD\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$
To US: - US = (double)(180/0xFFFFFFFE * LS) *To LS:* - LS = (int32)round(0xFFFFFFFE/180 * US)	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int32.

8.23.1 Example Frame Center Latitude

The center of the captured image or image sequence has a real earth coordinate represented by a latitude-longitude-altitude triplet. Frame centers that lie above the horizon typically do not correspond to a point on the earth (an example being the tracking of an airborne object) and should either not be reported, or be reported as an "error".

Note that the int32 used in the LS value is encoded using two's complement.

8.24 Tag 24: Frame Center Longitude Conversion

LS Tag	24	Units	Range	Format
LS Name	Frame Center Longitude	Degrees	+/- 180	int32
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Terrain Longitude of frame center. Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-180. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~84 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_24\_dec = \left(\frac{360}{4294967294} * LS\_24\right)$

Example Value	Example LS Packet
29.157890122923 Degrees	[K][L][V] = [0d24][0d4][0x14 BC 08 2B]

US Key	06 0E 2B 34 01 01 01 01 07 01 02 01 03 04 00 00	ESD Digraph	To
US Name	Frame Center Longitude	ESD Name	Target Longitude

Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes	Notes
- Specifies the video frame center point geographic location in decimal degrees of longitude. - Positive values indicate eastern hemisphere. - Negative values indicate western hemisphere.	- Longitude of the EO/IR payload's aimpoint on the ground. + Means East longitude. All longitude coordinates use WGS84.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$	$ESD\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$
To US: - US = (double)(360/0xFFFFFFFE * LS) *To LS:* - LS = (int32)round(0xFFFFFFFE/360 * US)	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int32.

8.24.1 Example Frame Center Longitude

The center of the captured image or image sequence has a real earth coordinate represented by a latitude-longitude-altitude triplet. Frame centers that lie above the horizon typically do not correspond to a point on the earth (an example being the tracking of an airborne object) and should either not be reported, or be reported as an “error”.

Note that the int32 used in the LS value is encoded using two’s complement.

8.25 Tag 25: Frame Center Elevation Conversion

LS Tag	25	Units	Range	Format
LS Name	Frame Center Elevation	Meters	-900..19000	uint16
US Mapped Key	06 0E 2B 34 01 01 01 0A 07 01 02 01 03 16 00 00			
Notes		Conversion Formula		
- Terrain elevation at frame center relative to Mean Sea Level (MSL). - Map 0..(2^16-1) to -900..19000 meters. - Resolution: ~0.3 meters.		$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right) - Offset$ $LS\_25\_dec = \left(\frac{19900}{65535} * LS\_25\right) - 900$		
Example Value		Example LS Packet		
3216.037 Meters		[K][L][V] = [0d25][0d2][0x34 F3]		

US Key	x		ESD Digraph	Te	
US Name	x		ESD Name	Frame Center Elevation	
Units	Range	Format	Units	Range	Format
x	x	x	Feet	+/- 0..99,999	PN
Notes			Notes		
- x			- Terrain elevation at frame center.		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			$ESD\_dec = \left(\frac{19900}{65535}*LS\_uint-900\right) * \frac{3.2808399ft}{1m}$ *To ESD:* - Convert LS to decimal. - Account for units. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Account for units. - Map decimal to uint16.		

8.25.1 Example Frame Center Elevation

For legacy purposes, both MSL (Tag 25) and HAE (Tag 78) representations of Frame Center Elevation MAY appear in the same ST 0601 packet. A single representation is preferred favoring the HAE version (Tag 78).

The center of the captured image or image sequence has a real earth coordinate represented by a latitude-longitude-altitude triplet. Frame centers that lie above the horizon typically do not correspond to a point on the earth (an example being the tracking of an airborne object) and should either not be reported, or be reported as an “error”.

The altitude is represented as height above mean sea level (MSL).

8.26 Tag 26: Offset Corner Latitude Point 1 Conversion

LS Tag	26	Units	Range	Format
LS Name	Offset Corner Latitude Point 1	Degrees	+/-0.075	int16
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Latitude, offset for upper left corner. Based on WGS84 ellipsoid. - Use with Frame Center Latitude. - Map -(2^15-1)..(2^15-1) to +/-0.075. - Use -(2^15) as an "error" indicator. - -(2^15) = 0x8000. - Resolution: ~1.2micro deg, ~0.25meters at equator.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right) + LS\_23\_dec$ $LS\_26\_dec = \left(\frac{0.15}{65534} * LS\_26\right) + LS\_23\_dec$

Example Value	Example LS Packet
-10.579637999887 Corrected Degrees	[K][L][V] = [0d26][0d2][0xC0 6E]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 07 01 00		ESD Digraph	Rg	
US Name	Corner Latitude Point 1 (Decimal Degrees)		ESD Name	SAR Latitude 4	
Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes	Notes
- Latitude coordinate of corner 1 of an image or bounding rectangle. - Positive (+) is northern hemisphere. - Negative (-) is southern hemisphere.	- The latitude of the upper left corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_23\_dec$	$ESD\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_23\_dec$
To US: - US = (double)((0.15/0xFFFE * LS) + LS_23_dec) *To LS:* - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LAT))	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.26.1 Example Corner Latitude Point 1

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair (Figure 8-12). Corner points that lie above the horizon typically do not correspond to a point on the earth, or corner points lying outside of the mapped range should either not be reported, or be reported as an “error”.

Corner point 1 is the upper left corner of the captured image as highlighted in red.

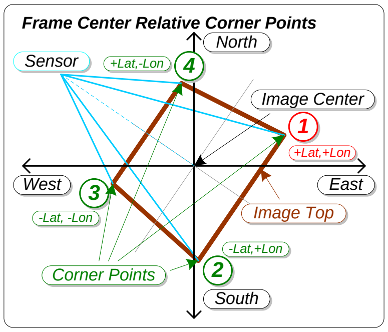

Figure 8-12: Offset Corner Point 1

The Offset Corner Latitude Point 1 is added to the Frame Center Latitude metadata item to determine the Latitude of the first corner point of a motion image. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated Motion Imagery corner point. Value is encoded using two's complement.

8.27 Tag 27: Offset Corner Longitude Point 1 Conversion

		Units	Range	Format
LS Tag	27	Degrees	+/-0.075	int16
LS Name	Offset Corner Longitude Point 1			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Longitude, offset for upper left corner. Based on WGS84 ellipsoid. - Use with Frame Center Longitude. - Map -(2^15-1)..(2^15-1) to +/-0.075. - Use -(2^15) as an "error" indicator. - -(2^15) = 0x8000. - Resolution: ~1.2micro deg, ~0.25meters at equator.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right) + LS\_24\_dec$ $LS\_27\_dec = \left(\frac{0.15}{65534} * LS\_27\right) + LS\_24\_dec$

Example Value	Example LS Packet
29.1273677986333 Corrected Degrees	[K][L][V] = [0d27][0d2][0xCB E9]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0B 01 00	ESD Digraph	Rh
US Name	Corner Longitude Point 1 (Decimal Degrees)	ESD Name	SAR Longitude 4

Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes	Notes
- Longitude coordinate of corner 1 of an image or bounding rectangle. - Positive (+) is eastern hemisphere. - Negative (-) is western hemisphere.	- The longitude of the upper left corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_24\_dec$	$ESD\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_24\_dec$
To US: - US = (double)((0.15/0xFFFE * LS) + LS_24_dec) *To LS:* - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LON))	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.27.1 Example Corner Longitude Point 1

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, or corner points lying outside of the mapped range, should either not be reported, or be reported as an “error”.

Corner point 1 is the upper left corner of the captured image. See Figure 8-12 for Tag 26 above.

The Offset Corner Longitude Point 1 is added to the Frame Center Longitude metadata item to determine the Longitude of the first corner point of a motion image. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated Motion Imagery corner point. Value is encoded using two’s complement.

8.28 Tag 28: Offset Corner Latitude Point 2 Conversion

LS Tag	28	Units	Range	Format
LS Name	Offset Corner Latitude Point 2	Degrees	+/-0.075	int16
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Latitude, offset for upper right corner. Based on WGS84 ellipsoid. - Use with Frame Center Latitude. - Map -(2^15-1)..(2^15-1) to +/-0.075. - Use -(2^15) as an "error" indicator. - -(2^15) = 0x8000. - Resolution: ~1.2micro deg, ~0.25meters at equator.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right) + LS\_23\_dec$ $LS\_28\_dec = \left(\frac{0.15}{65534} * LS\_28\right) + LS\_23\_dec$

Example Value	Example LS Packet
-10.5661816260963 Corrected Degrees	[K][L][V] = [0d28][0d2][0xD7 65]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 08 01 00	ESD Digraph	Ra
US Name	Corner Latitude Point 2 (Decimal Degrees)	ESD Name	SAR Latitude 1

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes (US)	Notes (ESD)
- Latitude coordinate of corner 2 of an image or bounding rectangle. - Positive (+) is northern hemisphere. - Negative (-) is southern hemisphere.	- The latitude of the upper right corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_23\_dec$	$ESD\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_23\_dec$
To US: - US = (double)((0.15/0xFFFE * LS) + LS_23_dec) *To LS:* - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LAT))	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.28.1 Example Corner Latitude Point 2

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, or corner points lying outside of the mapped range, should either not be reported, or be reported as an "error".

Corner point 2 is the upper right corner of the captured image as highlighted in red (Figure 8-13).

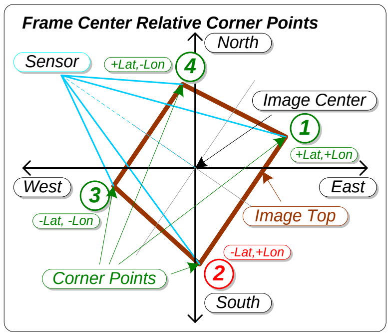

Figure 8-13: Offset Corner Point 2

The Offset Corner Latitude Point 2 is added to the Frame Center Latitude metadata item to determine the Latitude of the second corner point of a motion image. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated Motion Imagery corner point. Value is encoded using two's complement.

8.29 Tag 29: Offset Corner Longitude Point 2 Conversion

		Units	Range	Format
LS Tag	29	Degrees	+/-0.075	int16
LS Name	Offset Corner Longitude Point 2			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Longitude, offset for upper right corner. Based on WGS84 ellipsoid. - Use with Frame Center Longitude. - Map -(2^15-1)..(2^15-1) to +/-0.075. - Use -(2^15) as an "error" indicator. - -(2^15) = 0x8000. - Resolution: ~1.2micro deg, ~0.25meters at equator.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right) + LS\_24\_dec$ $LS\_29\_dec = \left(\frac{0.15}{65534} * LS\_29\right) + LS\_24\_dec$

Example Value	Example LS Packet
29.140824172424 Corrected Degrees	[K][L][V] = [0d29][0d2][0xE2 E0]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0C 01 00	ESD Digraph	Rb
US Name	Corner Longitude Point 2 (Decimal Degrees)	ESD Name	SAR Longitude 1

Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes	Notes
- Longitude coordinate of corner 2 of an image or bounding rectangle. - Positive (+) is eastern hemisphere. - Negative (-) is western hemisphere.	- The longitude of the upper right corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_24\_dec$	$ESD\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_24\_dec$
To US: - US = (double)((0.15/0xFFFE * LS) + LS_24_dec) *To LS:* - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LON))	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.29.1 Example Corner Longitude Point 2

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, or corner points lying outside of the mapped range, should either not be reported, or be reported as an "error".

Corner point 2 is the upper right corner of the captured image. See Figure 8-13 for Tag 28 above.

The Offset Corner Longitude Point 2 is added to the Frame Center Longitude metadata item to determine the Longitude of the second corner point of a motion image. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated Motion Imagery corner point. Value is encoded using two's complement.

8.30 Tag 30: Offset Corner Latitude Point 3 Conversion

		Units	Range	Format
LS Tag	30	Degrees	+/-0.075	int16
LS Name	Offset Corner Latitude Point 3			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Latitude, offset for lower right corner. Based on WGS84 ellipsoid. - Use with Frame Center Latitude. - Map -(2^15-1)..(2^15-1) to +/-0.075. - Use -(2^15) as an "error" indicator. - -(2^15) = 0x8000. - Resolution: ~1.2micro deg, ~0.25meters at equator.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right) + LS\_23\_dec$ $LS\_30\_dec = \left(\frac{0.15}{65534} * LS\_30\right) + LS\_23\_dec$

Example Value	Example LS Packet
-10.5527275411938 Corrected Degrees	[K][L][V] = [0d30][0d2][0xEE 5B]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 09 01 00	ESD Digraph	Rc
US Name	Corner Latitude Point 3 (Decimal Degrees)	ESD Name	SAR Latitude 2

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes	Notes
- Latitude coordinate of corner 3 of an image or bounding rectangle. - Positive (+) is northern hemisphere. - Negative (-) is southern hemisphere.	- The latitude of the lower right corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_23\_dec$	$ESD\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_23\_dec$
To US: - US = (double)((0.15/0xFFFE * LS) + LS_23_dec) *To LS:* - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LAT))	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.30.1 Example Corner Latitude Point 3

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, or corner points lying outside of the mapped range, should either not be reported, or be reported as an “error”.

Corner point 3 is the lower right corner of the captured image as highlighted in red (seeFigure 8-14).

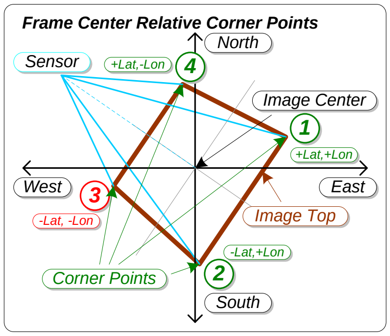

Figure 8-14: Offset Corner Point 3

The Offset Corner Latitude Point 3 is added to the Frame Center Latitude metadata item to determine the Latitude of the third corner point of a motion image. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated Motion Imagery corner point. Value is encoded using two's complement.

8.31 Tag 31: Offset Corner Longitude Point 3 Conversion

LS Tag	31	Units	Range	Format
LS Name	Offset Corner Longitude Point 3	Degrees	+/-0.075	int16
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Longitude, offset for lower right corner. Based on WGS84 ellipsoid. - Use with Frame Center Longitude. - Map -(2^15-1)..(2^15-1) to +/-0.075. - Use -(2^15) as an "error" indicator. - -(2^15) = 0x8000. - Resolution: ~1.2micro deg, ~0.25meters at equator.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right) + LS\_24\_dec$ $LS\_31\_dec = \left(\frac{0.15}{65534} * LS\_31\right) + LS\_24\_dec$

Example Value	Example LS Packet
29.1542782573265 Corrected Degrees	[K][L][V] = [0d31][0d2][0xF9 D6]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0D 01 00	ESD Digraph	Rd
US Name	Corner Longitude Point 3 (Decimal Degrees)	ESD Name	SAR Longitude 2

Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes	Notes
- Longitude coordinate of corner 3 of an image or bounding rectangle. - Positive (+) is eastern hemisphere. - Negative (-) is western hemisphere.	- The longitude of the lower right corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_24\_dec$	$ESD\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_24\_dec$
To US: - US = (double)((0.15/0xFFFE * LS) + LS_24_dec) *To LS:* - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LON))	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.31.1 Example Corner Longitude Point 3

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, or corner points lying outside of the mapped range, should either not be reported, or be reported as an “error”.

Corner point 3 is the lower right corner of the captured image. See Figure 8-14 for Tag 30 above.

The Offset Corner Longitude Point 3 is added to the Frame Center Longitude metadata item to determine the Longitude of the third corner point of a motion image. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated Motion Imagery corner point. Value is encoded using two’s complement.

8.32 Tag 32: Offset Corner Latitude Point 4 Conversion

		Units	Range	Format
LS Tag	32	Degrees	+/-0.075	int16
LS Name	Offset Corner Latitude Point 4			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Latitude, offset for lower left corner. Based on WGS84 ellipsoid. - Use with Frame Center Latitude. - Map -(2^15-1)..(2^15-1) to +/-0.075. - Use -(2^15) as an "error" indicator. - -(2^15) = 0x8000. - Resolution: ~1.2micro deg, ~0.25meters at equator.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right) + LS\_23\_dec$ $LS\_32\_dec = \left(\frac{0.15}{65534} * LS\_32\right) + LS\_23\_dec$

Example Value	Example LS Packet
-10.5392711674031 Corrected Degrees	[K][L][V] = [0d32][0d2][0x05 52]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0A 01 00	ESD Digraph	Re
US Name	Corner Latitude Point 4 (Decimal Degrees)	ESD Name	SAR Latitude 3

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes	Notes
- Latitude coordinate of corner 4 of an image or bounding rectangle. - Positive (+) is northern hemisphere. - Negative (-) is southern hemisphere.	- The latitude of the lower left corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_23\_dec$	$ESD\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_23\_dec$
To US: - US = (double)((0.15/0xFFFE * LS) + LS_23_dec) *To LS:* - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LAT))	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.32.1 Example Corner Latitude Point 4

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, or corner points lying outside of the mapped range, should either not be reported, or be reported as an “error”.

Corner point 4 is the lower left corner of the captured image as highlighted in red (see Figure 8-15).

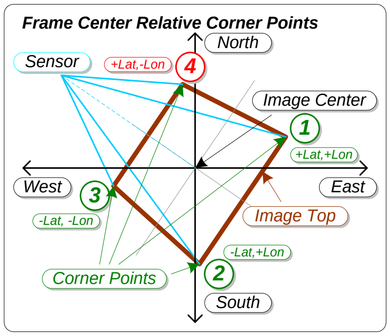

Figure 8-15: Offset Corner Point 4

The Offset Corner Latitude Point 4 is added to the Frame Center Latitude metadata item to determine the Latitude of the fourth corner point of a motion image. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated Motion Imagery corner point. Value is encoded using two's complement.

8.33 Tag 33: Offset Corner Longitude Point 4 Conversion

		Units	Range	Format
LS Tag	33	Degrees	+/-0.075	int16
LS Name	Offset Corner Longitude Point 4			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Longitude, offset for lower left corner. Based on WGS84 ellipsoid. - Use with Frame Center Longitude. - Map -(2^15-1)..(2^15-1) to +/-0.075. - Use -(2^15) as an "error" indicator. - -(2^15) = 0x8000. - Resolution: ~1.2micro deg, ~0.25meters at equator.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right) + LS\_24\_dec$ $LS\_33\_dec = \left(\frac{0.15}{65534} * LS\_33\right) + LS\_24\_dec$

Example Value	Example LS Packet
29.1677346311172 Corrected Degrees	[K][L][V] = [0d33][0d2][0x10 CD]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0E 01 00	ESD Digraph	Rf
US Name	Corner Longitude Point 4 (Decimal Degrees)	ESD Name	SAR Longitude 3

Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes	Notes
- Longitude coordinate of corner 4 of an image or bounding rectangle. - Positive (+) is eastern hemisphere. - Negative (-) is western hemisphere.	- The longitude of the lower left corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_24\_dec$	$ESD\_dec = \left(\frac{0.15}{65534} * LS\_int\right) + LS\_24\_dec$
To US: - US = (double)((0.15/0xFFFE * LS) + LS_24_dec) *To LS:* - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LON))	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int16.

8.33.1 Example Corner Longitude Point 4

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, or corner points lying outside of the mapped range, should either not be reported, or be reported as an “error”.

Corner point 4 is the lower left corner of the captured image. See Figure 8-15 for Key 32 above.

The Offset Corner Longitude Point 4 is added to the Frame Center Longitude metadata item to determine the Longitude of the fourth corner point of a motion image. Both KLV data items must be converted to decimal prior to addition to determine the actual measured or calculated Motion Imagery corner point. Value is encoded using two’s complement.

8.34 Tag 34: Icing Detected Conversion

LS Tag	34	Units	Range	Format
LS Name	Icing Detected	Icing Code	0..255	uint8
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 0C 00 00 00			

Notes	Conversion Formula
- Flag for icing detected at aircraft location. - 0: Detector off - 1: No icing Detected - 2: Icing Detected	x x

Example Value	Example LS Packet
Invalid Icing Code	[K][L][V] = [0d34][0d1][0x9B]

US Key	x		ESD Digraph	Id	
US Name	x		ESD Name	Icing Detected	
Units	Range	Format	Units	Range	Format
x	x	x	Icing Code	0..2	N

Notes	Notes
- x	- Output of the aircrafts icing detector - 0: Detector off - 1: No icing detected - 2: Icing detected

US Conversion	ESD Conversion
x	x
To US: - x	*To ESD:* - Convert string to ID code.
To LS: - x	*To LS:* - Convert ID code to string.

8.34.1 Example Icing Detected

This metadata item signals when the icing sensor detects water forming on its vibrating probe.

8.35 Tag 35: Wind Direction Conversion

LS Tag	35	Units	Range	Format
LS Name	Wind Direction	Degrees	0..360	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 0D 00 00 00			

Notes	Conversion Formula
- Wind direction at aircraft location. This is the direction the wind is coming from relative to true north. - Map 0..(2^16-1) to 0..360. - Resolution: ~5.5 milli degrees.	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_35\_dec = \left(\frac{360}{65535} * LS\_35\right)$

Example Value	Example LS Packet
235.924 Degrees	[K][L][V] = [0d35][0d2][0xA7 C4]

US Key	x		ESD Digraph	Wd	
US Name	x		ESD Name	Wind Direction	
Units	Range	Format	Units	Range	Format
x	x	x	Degrees	0..359	DDD

Notes	Notes
- x	- Direction (from North) from which the wind is blowing at the aircraft location

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	$ESD\_dec = \left(\frac{360}{65534} * LS\_uint\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Map decimal to uint16.

8.35.1 Example Wind Direction

The direction the air body around the aircraft is coming from relative to true north.

8.36 Tag 36: Wind Speed Conversion

LS Tag	36	Units	Range	Format
LS Name	Wind Speed	Meters/Second	0..100	uint8
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 0E 00 00 00			

Notes
- Wind speed at aircraft location.
- Map 0..255 to 0..100 meters/second.
- 1 m/s = 1.94384449 knots.
- Resolution: ~0.4 meters / second.

Conversion Formula

$$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$$

$$LS\_36\_dec = \left(\frac{100}{255} * LS\_36\right)$$

Example Value	Example LS Packet
69.80392 m/s	[K][L][V] = [0d36][0d1][0xB2]

US Key	x		ESD Digraph	Ws	
US Name	x		ESD Name	Wind Speed	
Units	Range	Format	Units	Range	Format
x	x	x	Knots	0..99	NN

Notes (US)
- x

Notes (ESD)
- Wind Speed (relative to the Earth) at the aircraft location.

US Conversion

x

To US:
- x

To LS:
- x

ESD Conversion

$$ESD\_dec = \left(\frac{100}{255} * LS\_uint\right) * \frac{1.94384449knots}{1m/s}$$

To ESD:
- Convert LS to decimal.
- Account for units.
- Convert knots to ASCII.

To LS:
- Convert ESD ASCII to decimal.
- Account for units.
- Convert meters to uint8.

8.36.1 Example Wind Speed

The speed of the body of air that surrounds the aircraft relative to the ground is captured in this wind speed metadata item.

8.37 *Tag 37: Static Pressure Conversion*

LS Tag	37	Units	Range	Format
LS Name	Static Pressure	Millibar	0..5000	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 0F 00 00 00			

Notes	Conversion Formula
- Static pressure at aircraft location. - Map 0..(2^16-1) to 0..5000 mbar. - 1 mbar = 0.0145037738 PSI. - Resolution: ~0.08 Millibar	$LS\_dec = \left(\frac{LS\ range}{uint\_range} * LS\_uint\right)$ $LS\_37\_dec = \left(\frac{5000}{65535} * LS\_37\right)$

Example Value	Example LS Packet
3725.185 mbar	[K][L][V] = [0d37][0d2][0xBE BA]

US Key	x		ESD Digraph	Ps	
US Name	x		ESD Name	Static Pressure	
Units	Range	Format	Units	Range	Format
x	x	x	PSI	0..99.99	DD.HH

Notes	Notes
- x	- Static Pressure

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	$ESD\_dec = \left(\frac{5000}{65535} * LS\_uint\right) * \frac{0.0145037738PSI}{1mbar}$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Map decimal to uint16.

8.37.1 Example Static Pressure

The static pressure is the pressure of the air that surrounds the aircraft. Static pressure is measured by a sensor mounted out of the air stream on the side of the fuselage. This is used with impact pressure to compute indicated airspeed, true airspeed, and density altitude.

8.38 Tag 38: Density Altitude Conversion

LS Tag	38	Units	Range	Format
LS Name	Density Altitude	Meters	-900..19000	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 10 00 00 00			

Notes

- Density altitude at aircraft location. Relative aircraft performance metric based on outside air temperature, static pressure, and humidity.
- Map 0..(2^16-1) to -900..19000 meters.
- Offset = -900.
- 1 meter = 3.2808399 feet.
- Resolution: ~0.3 meters.

Conversion Formula

$$LS\_dec = \left(\frac{LS\ range}{uint\_range} * LS\_uint\right) - Offset$$

$$LS\_38\_dec = \left(\frac{19900}{65535} * LS\_38\right) - 900$$

Example Value	Example LS Packet
14818.68 Meters	[K][L][V] = [0d38][0d2][0xCA 35]

US Key	x	ESD Digraph	Da
US Name	x	ESD Name	Density Altitude

Units	Range	Format	Units	Range	Format
x	x	x	Feet	+/- 99,999	PN

Notes (US)

- x

Notes (ESD)

- Density Altitude of the aircraft.

US Conversion

x

To US:

- x

To LS:

- x

ESD Conversion

$$ESD\_dec = \left(\frac{19900}{65535} * LS\_uint - 900\right) * \frac{3.2808399ft}{1m}$$

To ESD:

- Convert LS to decimal.
- Account for units.
- Convert decimal to ASCII.

To LS:

- Convert ESD ASCII to decimal.
- Account for units.
- Map decimal to uint16.

8.38.1 Example Density Altitude

Density altitude is the pressure altitude corrected for non-standard temperature variation. Density altitude is a relative metric of the takeoff, climb, and other performance related parameters of an aircraft.

8.39 Tag 39: Outside Air Temperature Conversion

		Units	Range	Format
LS Tag	39	Celcius	-128..+127	int8
LS Name	Outside Air Temperature			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 11 00 00 00			

Notes	Conversion Formula
- Temperature outside of aircraft. - -128..127 Degrees Celsius. - Resolution: 1 degree celsius.	LS_dec = LS_int LS_39_dec = LS_39

Example Value	Example LS Packet
84 Celcius	[K][L][V] = [0d39][0d1][0x54]

US Key	x		ESD Digraph	At	
US Name	x		ESD Name	Air Temperature	
Units	Range	Format	Units	Range	Format
x	x	x	Celcius	+/- 99	PDD

Notes	Notes
- x	- Outside air temperature measured at the aircraft

US Conversion	ESD Conversion
x	ESD_dec = LS_int
To US: - x	*To ESD:* - Convert int8 to string.
To LS: - x	*To LS:* - Convert string to int8.

8.39.1 Example Outside Air Temperature

The measured temperature outside of the platform is captured in the outside air temperature metadata item.

Note that the value is encoded using two's complement.

8.40 Tag 40: Target Location Latitude Conversion

LS Tag	40	**Units**	**Range**	**Format**
LS Name	Target Location Latitude	Degrees	+/- 90	int32
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 03 02 00 00 00			
Notes		**Conversion Formula**		
- Calculated Target latitude. This is the crosshair location if different from frame center. - Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-90. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~42 nano degrees.		$LS\_dec = \left(\frac{LS\ range}{int\_range} * LS\_int\right)$ $LS\_40\_dec = \left(\frac{180}{4294967294} * LS\_40\right)$		
Example Value	**Example LS Packet**			
-79.1638500518929 Degrees	[K][L][V] = [0d40][0d4][0x8F 69 52 62]			

8.40.1 Example Target Location Latitude

The crosshair or target location of a captured image or image sequence has a real earth coordinate represented by a latitude-longitude-elevation triplet and may differ from the center of the captured image. Target locations that lie above the horizon do not correspond to a point on the earth and should either not be reported, or be reported as an “error”.

Note that the int32 used in the LS value is encoded using two’s complement.

8.41 Tag 41: Target Location Longitude Conversion

		Units	Range	Format
LS Tag	41	Degrees	+/-180	int32
LS Name	Target Location Longitude			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 03 03 00 00 00			

Notes	Conversion Formula
- Calculated Target longitude. This is the crosshair location if different from frame center. - Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-180. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~84 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_41\_dec = \left(\frac{360}{4294967294} * LS\_41\right)$

Example Value	Example LS Packet
166.400812960416 Degrees	[K][L][V] = [0d41][0d4][0x76 54 57 F2]

8.41.1 Example Target Location Longitude

The crosshair or target location of a captured image or image sequence has a real earth coordinate represented by a latitude-longitude-elevation triplet and may differ from the center of the captured image. Target locations that lie above the horizon do not correspond to a point on the earth and should either not be reported, or be reported as an “error”.

Note that the int32 used in the LS value is encoded using two’s complement.

8.42 Tag 42: Target Location Elevation Conversion

LS Tag	42	Units	Range	Format
LS Name	Target Location Elevation	Meters	-900..19000	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 03 04 00 00 00			

Notes	Conversion Formula
- Calculated target elevation. This is the crosshair location if different from frame center. - Map 0..(2^16-1) to -900..19000 meters. - Offset = -900. - 1 meter = 3.2808399 feet. - Resolution: ~0.3 meters.	$LS\_dec = \left(\frac{LS\ range}{uint\_range} * LS\_uint\right) - Offset$ $LS\_42\_dec = \left(\frac{19900}{65535} * LS\_42\right) - 900$

Example Value	Example LS Packet
18389.05 Meters	[K][L][V] = [0d42][0d2][0xF8 23]

8.42.1 Example Target Location Elevation

The crosshair or target location of a captured image or image sequence has a real earth coordinate represented by a latitude-longitude-elevation triplet and may differ from the center of the captured image. Target locations that lie above the horizon do not correspond to a point on the earth and should either not be reported, or be reported as an “error”.

8.43 Tag 43: Target Track Gate Width Conversion

LS Tag	43	Units	Range	Format
LS Name	Target Track Gate Width	Pixels	0..512	uint8
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 03 05 00 00 00			
Notes		Conversion Formula		
- Tracking gate width (x value) of tracked target within field of view. - Closely tied to source video resolution in pixels.		LS_dec = 2 * LS_uint LS_43_dec = round(2 * LS_43)		
Example Value	Example LS Packet			
6 Pixels	[K][L][V] = [0d43][0d1][0x03]			

8.43.1 Example Target Track Gate Width

The target track gate width is used with Target Tracking Sensors that specify the pixel width of a tracking gate to be displayed about a target location.

8.44 Tag 44: Target Track Gate Height Conversion

		Units	Range	Format
LS Tag	44	Pixels	0..512	uint8
LS Name	Target Track Gate Height			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 03 06 00 00 00			

Notes	Conversion Formula
- Tracking gate height (y value) of tracked target within field of view. - Closely tied to source video resolution in pixels.	LS_dec = 2 * LS_uint LS_44_dec = round(2 * LS_44)

Example Value	Example LS Packet
30 Pixels	[K][L][V] = [0d44][0d1][0x0F]

8.44.1 Example Target Track Gate Height

The target track gate height is used with Target Tracking Sensors that specify the pixel height of a tracking gate to be displayed about a target location.

8.45 Tag 45: Target Error Estimate - CE90 Conversion

		Units	Range	Format
LS Tag	45	Meters	0..4095	uint16
LS Name	Target Error Estimate - CE90			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 03 07 00 00 00			

Notes	Conversion Formula
- Circular Error 90 (CE90) is the estimated error distance in the horizontal direction. - Specifies the radius of 90% probability on a plane tangent to the earth's surface. - Res: ~0.0624 meters	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint \right)$ $LS\_45\_dec = \left(\frac{4095}{65535} * LS\_45 \right)$

Example Value	Example LS Packet
425.319 Meters	[K][L][V] = [0d45][0d2][0x1A 95]

8.45.1 Example Target Error Estimate – Circular Error 90% (CE90)

Target covariance values are represented in an easting-northing-up coordinate system centered about the target point. This is shown below (refer to Figure 8-16):

Covariance Matrix:

$$Q = \begin{bmatrix} \sigma_e^2 & \sigma_{en} & \sigma_{eu} \\ \sigma_{ne} & \sigma_n^2 & \sigma_{nu} \\ \sigma_{ue} & \sigma_{un} & \sigma_u^2 \end{bmatrix}$$

Min and Max Sigma Values:

$$\sigma_{max}^2 = \frac{\left(\sigma_e^2 + \sigma_n^2\right) + \sqrt{\left(\sigma_e^2 + \sigma_n^2\right)^2 - 4\left(\sigma_e^2\sigma_n^2 - \sigma_{en}^2\right)}}{2}$$

$$\sigma_{min}^2 = \frac{\left(\sigma_e^2 + \sigma_n^2\right) - \sqrt{\left(\sigma_e^2 + \sigma_n^2\right)^2 - 4\left(\sigma_e^2\sigma_n^2 - \sigma_{en}^2\right)}}{2}$$

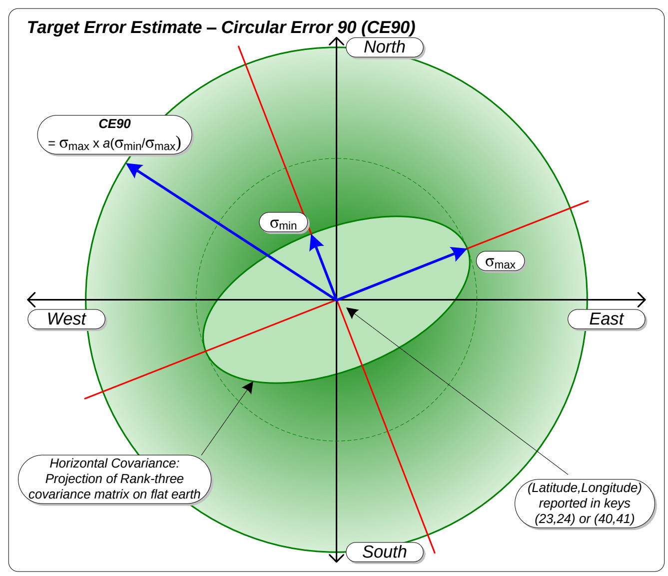

Figure 8-16: Target Error Estimate - Circular Error 90%

CE90 represents the 90 percent probability circular error radius of absolute horizontal accuracy. With $\sigma_{\max}$ and $\sigma_{\min}$ known, the Circular Error for 90% confidence can be calculated as:

$CE90 = \sigma_{\max} \cdot a\left(\frac{\sigma_{\min}}{\sigma_{\max}}\right)$ where $a(x) = 0.4194x^2 + 0.0774x + 1.648$. This is one means for determining CE90 from statistical data in the easting-northing-up coordinate system, yet similar calculations are allowed.

8.46 Tag 46: Target Error Estimate - LE90 Conversion

LS Tag	46	Units	Range	Format
LS Name	Target Error Estimate - LE90	Meters	0..4095	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 03 08 00 00 00			
Notes		**Conversion Formula**		
- Lateral Error 90 (LE90) is the estimated error distance in the vertical (or lateral) direction. - Specifies the interval of 90% probability in the local vertical direction. - Res: 0.0625 meters		LS_dec = (LS_range / uint_range * LS_uint) LS_46_dec = (4095 / 65535 * LS_46)		
Example Value	**Example LS Packet**			
609.0718 Meters	[K][L][V] = [0d46][0d2][0x26 11]			

8.46.1 Example Target Error Estimate – Linear Error 90% (LE90)

Target covariance values are represented in an easting-northing-up coordinate system centered about the target point. This is shown below:

Covariance Matrix:

$$Q = \begin{bmatrix} \sigma_e^2 & \sigma_{en} & \sigma_{eu} \\ \sigma_{ne} & \sigma_n^2 & \sigma_{nu} \\ \sigma_{ue} & \sigma_{un} & \sigma_u^2 \end{bmatrix}$$

Min and Max Sigma Values:

$$\sigma_{max}^2 = \frac{\left(\sigma_e^2 + \sigma_n^2\right) + \sqrt{\left(\sigma_e^2 + \sigma_n^2\right)^2 - 4\left(\sigma_e^2\sigma_n^2 - \sigma_{en}^2\right)}}{2}$$

$$\sigma_{min}^2 = \frac{\left(\sigma_e^2 + \sigma_n^2\right) - \sqrt{\left(\sigma_e^2 + \sigma_n^2\right)^2 - 4\left(\sigma_e^2\sigma_n^2 - \sigma_{en}^2\right)}}{2}$$

LE90 represents the 90 percent probability linear error of absolute vertical accuracy.

With the vertical (or “up”) variance known (σ_u), the 90 percent linear error can be calculated as $LE90 = 1.645 \cdot \sigma_u$. This is one means for determining LE90 from statistical data in the easting-northing-up coordinate system, yet similar calculations are allowed.

8.47 Tag 47: Generic Flag Data 01 Conversion

		Units	Range	Format
LS Tag	47	None	uint8	uint8
LS Name	Generic Flag Data 01			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 03 01 00 00 00			

Notes	Conversion Formula
- Generic Flagged Metadata - Position Format msb8..1lsb - 1- Laser Range 1on,0off - 2- Auto-Track 1on,0off - 3- IR Polarity 1blk,0wht - 4- Icing detected 1ice,0(off/no ice) - 5- Slant Range 1measured, 0calc - 6- Image Invalid 1invalid, 0valid - 7,8- Use 0	x x

Example Value	Example LS Packet
49	[K][L][V] = [0d47][0d1][0x31]

8.47.1 Example Generic Flag Data 01

Miscellaneous yes / no aircraft and image related data items are logged within the Generic Flag Data 01 metadata item.

Updates in ST 0601.3 include an indication (bit 5) that Slant Range (Tag 21) is either “calculated” (0) or “measured” (1).

Updates in ST 0601.5 include the Image Invalid flag (bit 6). This flag indicates the state of the associated Motion Imagery as being “valid” (0) or “invalid” (1). An invalid (or unusable) image can be due to a lens change, bad focus, or other camera parameter which significantly degrades the image quality.

8.48 Tag 48: Security Local Metadata Set Conversion

LS Tag	48	Units	Range	Format
LS Name	Security Local Metadata Set	None	Set	Set
US Mapped Key	Use ST0102 US key for Local Sets.			

Notes	Conversion Formula
- Local set tag to include the ST0102 Local Set Security Metadata items within ST0601. Use the ST0102 Local Set Tags within the ST0601 tag 0d48. - The length field is the size of all ST0102 metadata items to be packaged within tag 0d48.	x x

Example Value	Example LS Packet
x	[K][L][V] = [0d48][0dx][x]

US Key	06 0E 2B 34 02 03 01 01 0E 01 03 03 02 00 00 00		ESD Digraph	x	
US Name	Security Local Metadata Set		ESD Name	x	
Units	Range	Format	Units	Range	Format
x	x	x	x	x	x

Notes	Notes
- x	- x

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	x *To ESD:* - x *To LS:* - x

8.48.1 Example Security Local set

Both Universal Set tags and Local Set tags are defined for KLV formatted security items in MISB ST 0102. When incorporated within ST 0601, multiple security metadata KLV Local Set triplets are allowed to be contained within the 0d48 UAS LS metadata item.

8.49 Tag 49: Differential Pressure Conversion

LS Tag	49	Units	Range	Format
LS Name	Differential Pressure	Millibar	0..5000	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 01 00 00 00			
Notes		Conversion Formula		
- Differential pressure at aircraft location. Measured as the Stagnation/impact/total pressure minus static pressure. - Map 0..(2^16-1) to 0..5000 mbar. - 1 mbar = 0.0145037738 PSI. - Res: ~0.08 mbar		LS_dec = (LS_range / uint_range * LS_uint) LS_49_dec = (5000 / 65535 * LS_49)		
Example Value	Example LS Packet			
1191.958 mbar	[K][L][V] = [0d49][0d2][0x3D 07]			

8.49.1 Example Differential Pressure

Differential pressure provides a method of calculating relative velocity of an item as it passes through a fluid, or conversely the velocity of a fluid as it passes by an item. Velocity can be determined by differential pressure by the following:

$$v_1 = \sqrt{\frac{2p_d}{\rho}}$$

where p_d is the measured differential pressure (p_d = impact pressure minus static pressure = $p_i - p_s$), and ρ is the density of the fluid outside the item.

8.50 Tag 50: Platform Angle of Attack Conversion

LS Tag	50	Units	Range	Format
LS Name	Platform Angle of Attack	Degrees	+/- 20	int16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 02 00 00 00			
Notes		Conversion Formula		
- Platform Attack Angle. Angle between platform longitudinal axis and relative wind. - Positive angles for upward relative wind. - Map -(2^15-1)..(2^15-1) to +/-20. - Use -(2^15) as an "out of range" indicator. - -(2^15) = 0x8000. - Res: ~610 micro degrees.		$LS\_dec = \left(\frac{LS\ range}{int\_range} * LS\_int\right)$ $LS\_50\_dec = \left(\frac{40}{65534} * LS\_50\right)$		
Example Value	Example LS Packet			
-8.670177 Degrees	[K][L][V] = [0d50][0d2][0xC8 83]			

8.50.1 Example Platform Angle of Attack

For legacy purposes, both range-restricted (Tag 50) and full-range (Tag 92) representations of Platform Angle of Attack MAY appear in the same ST 0601 packet. A single representation is preferred favoring the full-range version (Tag 92).

The angle of attack of an airborne platform is the angle formed between the relative wind and platform longitudinal axis (line made by the fuselage). Positive angles for wind with a relative upward component. Refer to Figure 8-17.

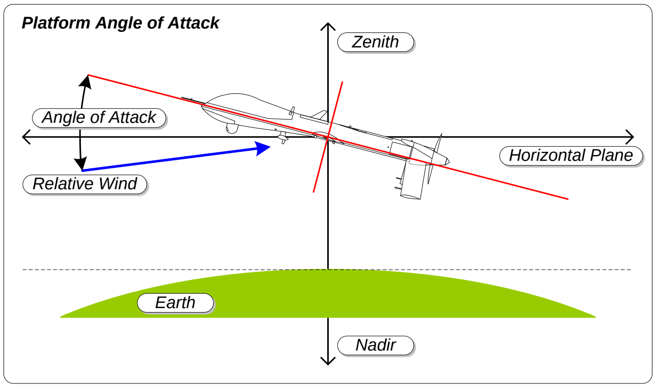

Figure 8-17: Platform Angle of Attack

Note that the int16 used in the LS value is encoded using two's complement.

8.51 Tag 51: Platform Vertical Speed Conversion

		Units	Range	Format
LS Tag	51	Meters/Second	+/- 180	int16
LS Name	Platform Vertical Speed			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 03 00 00 00			

Notes	Conversion Formula
- Vertical speed of the aircraft relative to zenith. Positive ascending, negative descending. - Map-(2^15-1)..(2^15-1) to +/-180 - Use -(2^15) as an "out of range" indicator. - -(2^15) = 0x8000. - Resolution: ~ 0.0055 meters/second.	$LS\_dec = \left(\frac{LS\ range}{int\_range} * LS\_int\right)$ $LS\_51 = \left(\frac{360}{65534} * LS\_51\right)$

Example Value	Example LS Packet
-61.88693 m/s	[K][L][V] = [0d51][0d2][0xD3 FE]

8.51.1 Example Vertical Speed

The vertical speed metadata item is the climb or decent rate in meters per second of an airborne platform in the zenith direction. Positive values indicate an ascending platform, while negative values indicate descending.

Note that the int16 used in the LS value is encoded using two's complement.

8.52 Tag 52: Platform Sideslip Angle Conversion

		Units	Range	Format
LS Tag	52	Degrees	+/- 20	int16
LS Name	Platform Sideslip Angle			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 04 00 00 00			

Notes	Conversion Formula
- The sideslip angle is the angle between the platform longitudinal axis and relative wind. - Positive angles to right wing, neg to left. - Map -(2^15-1)..(2^15-1) to +/-20. - Use -(2^15) as an "out of range" indicator. - -(2^15) = 0x8000. - Res: ~610 micro deg.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_52\_dec = \left(\frac{40}{65534} * LS\_52\right)$

Example Value	Example LS Packet
-5.082475 Degrees	[K][L][V] = [0d52][0d2][0xDF 79]

8.52.1 Example Platform Sideslip Angle

For legacy purposes, both range-restricted (Tag 52) and full-range (Tag 93) representations of Platform Sideslip Angle MAY appear in the same ST 0601 packet. A single representation is preferred favoring the full-range version (Tag 93).

The angle formed between the platform longitudinal axis (line made by the fuselage) and the relative wind is the sideslip angle. A negative sideslip angle is depicted in Figure 8-18:

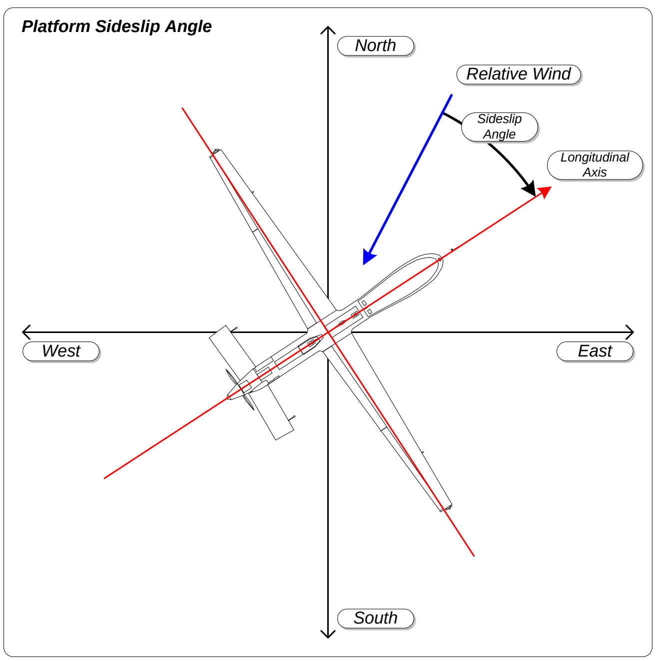

Figure 8-18: Platform Sideslip Angle

Note that the int16 used in the LS value is encoded using two's complement.

8.53 Tag 53: Airfield Barometric Pressure Conversion

		Units	Range	Format
LS Tag	53	Millibar	0..5000	uint16
LS Name	Airfield Barometric Pressure			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 02 02 00 00 00			

Notes	Conversion Formula
- Local pressure at airfield of known height. Pilot's responsibility to update. - Map 0..(2^16-1) to 0..5000 mbar. - 1013.25mbar = 29.92inHg - Min/max recorded values of 870/1086mbar. - Resolution: ~0.08 Millibar	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_53\_dec = \left(\frac{5000}{65535} * LS\_53\right)$

Example Value	Example LS Packet
2088.96 mbar	[K][L][V] = [0d53][0d2][0x6A F4]

8.53.1 Example Barometric Pressure at Airfield

Barometric pressure at airfield is used with altimeters to display airfield elevation when on the airfield.

8.54 Tag 54: Airfield Elevation Conversion

LS Tag LS Name US Mapped Key	54 Airfield Elevation 06 0E 2B 34 01 01 01 01 0E 01 01 02 03 00 00 00	Units	Range	Format
		Meters	-900..19000	uint16

Notes	Conversion Formula
- Elevation of Airfield corresponding to Airfield Barometric Pressure. - Map 0..(2^16-1) to -900..19000 meters. - Offset = -900. - 1 meter = 3.2808399 feet. - Resolution: ~0.3 meters.	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_int\right) - Offset$ $LS\_54\_dec = \left(\frac{19900}{65535} * LS\_54\right) - 900$

Example Value	Example LS Packet
8306.806 Meters	[K][L][V] = [0d54][0d2][0x76 70]

8.54.1 Example Airfield Elevation

Airfield elevation established at airfield location. This relates to the Barometric Pressure at Airfield metadata item.

8.55 *Tag 55: Relative Humidity Conversion*

LS Tag	55	Units	Range	Format
LS Name	Relative Humidity	Percent	0..100	uint8
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 09 00 00 00			
Notes		Conversion Formula		
- Relative Humidity at aircraft location. - Map 0..(2^8-1) to 0..100. - Resolution: ~0.4%.		$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_55\_dec = \left(\frac{100}{255} * LS\_55\right)$		
Example Value	Example LS Packet			
50.58823%	[K][L][V] = [0d55][0d1][0x81]			

8.55.1 Example Relative Humidity

Relative humidity is the ratio between the water vapor density and the saturation point of water vapor density and is expressed here as a percentage.

8.56 Tag 56: Platform Ground Speed Conversion

LS Tag	56	Units	Range	Format
LS Name	Platform Ground Speed	Meters/Second	0..255	uint8
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 05 00 00 00			
Notes		Conversion Formula		
- Speed projected to the ground of an airborne platform passing overhead. - 0..255 meters/sec. - 1 m/s = 1.94384449 knots. - Resolution: 1 meter/second.		LS_dec = LS_int LS_56_dec = round(LS_56)		
Example Value		Example LS Packet		
140 m/s		[K][L][V] = [0d56][0d1][0x8C]		

US Key	x	ESD Digraph	Gv	
US Name	x	ESD Name	Platform Ground Speed	

Units	Range	Format	Units	Range	Format
x	x	x	Knots	0..999	N

Notes	Notes
- x	- Speed on the ground of an airborne platform passing overhead.

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	ESD_dec = LS_uint *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Map decimal to uint16.

8.56.1 Example Platform Ground Speed

The ground speed of an airborne platform is the aircraft's speed as projected onto the ground.

8.57 Tag 57: Ground Range Conversion

LS Tag	57	**Units**	**Range**	**Format**
LS Name	Ground Range	Meters	0..5,000,000	uint32
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 06 00 00 00			

Notes	Conversion Formula
- Horizontal distance from ground position of aircraft relative to nadir, and target of interest. Dependent upon Slant Range and Depression Angle. - Map 0..(2^32-1) to 0..5000000 meters. - 1 nautical mile (knot) = 1852 meters. - Resolution: ~1.2 milli meters.	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_57\_dec = \left(\frac{5000000}{4294967295} * LS\_57\right)$

Example Value	Example LS Packet
3506979 Meters	[K][L][V] = [0d57][0d4][0xB3 8E AC F1]

US Key	x		ESD Digraph	Gr	
US Name	x		ESD Name	Ground Range	
Units	**Range**	**Format**	**Units**	**Range**	**Format**
x	x	x	Nautical Miles	0..18.00	II.HH

Notes	Notes
- x	- Horizontal distance between the sensor and the target. Measured in Nautical Miles.

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	$ESD\_dec = \left(\frac{5000000}{4294967295} * LS\_uint\right) * \frac{1852knot}{1m}$ *To ESD:* - Convert LS to decimal. - Account for units. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Account for units. - Convert ASCII to uint32.

8.57.1 Example Ground Range

Ground range is the horizontal distance between the aircraft/sensor location and the target of interest and does not account for terrain undulations.

8.58 Tag 58: Platform Fuel Remaining Conversion

		Units	Range	Format
LS Tag	58	Kilogram	0..10,000	uint16
LS Name	Platform Fuel Remaining			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 07 00 00 00			

Notes	Conversion Formula
- Remaining fuel on airborne platform. Metered as fuel weight remaining. - Map 0..(2^16-1) to 0..10000 Kilograms. - 1 kilogram = 2.20462262 pounds. - Resolution: ~.16 kilograms.	$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_58\_dec = \left(\frac{10000}{65535} * LS\_58\right)$

Example Value	Example LS Packet
6420.539 kg	[K][L][V] = [0d58][0d2][0xA4 5D]

US Key	x	ESD Digraph	Fr
US Name	x	ESD Name	Platform Fuel Remaining

Units	Range	Format	Units	Range	Format
x	x	x	Pounds	0..99,999	N

Notes	Notes
- x	- Remaining fuel on airborne platform. Metered as fuel weight remainging.

US Conversion	ESD Conversion
x *To US:* - x *To LS:* - x	$ESD\_dec = \left(\frac{10000}{65535} * LS\_uint\right) * \frac{2.04622621bs}{1kg}$ *To ESD:* - Convert LS to decimal. - Account for units. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Account for units. - Map decimal to uint16.

8.58.1 Example Platform Fuel Remaining

Platform fuel remaining indicates the current weight of fuel present on the host platform and is measured in kilograms.

8.59 Tag 59: Platform Call Sign Conversion

LS Tag	59	Units	Range	Format
LS Name	Platform Call Sign	String	1..127	ISO 646
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 04 01 01 00 00 00			
Notes		Conversion Formula		
- Call Sign of platform or operating unit. - Value field is Free Text.		x x		
Example Value	Example LS Packet			
TOP GUN	[K][L][V] = [0d59][0d7][0x54 4F 50 20 47 55 4E]			

US Key	x		ESD Digraph	Cs	
US Name	x		ESD Name	Platform Call Sign	
Units	Range	Format	Units	Range	Format
x	x	x	String	0..9	N
Notes			Notes		
- x			- First nine characters of the Call Sign of a group or squadron.		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			x *To ESD:* - Truncate LS String and convert to ESD *To LS:* - Convert ESD string to LS		

8.59.1 Example Platform Call Sign

The platform call sign is used to distinguish groups or squadrons of platforms within different operating units from one another. Call sign is often related to the aircraft tail number.

8.60 Tag 60: Weapon Load Conversion

		Units	Range	Format
LS Tag	60	uint16	x	nibble
LS Name	Weapon Load			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 12 00 00 00			

Notes	Conversion Formula
- Current weapons stored on aircraft broken into two bytes: - [K][L][V] = [0x41][0x02][[byte1][byte2]] - [byteN] = [[nib1][nib2]], nib1= msn - byte1-nib1 = Station Number - byte1-nib2 = Substation Number - byte2-nib1 = Weapon Type - byte2-nib2 = Weapon Variant	x x

Example Value	Example LS Packet
45016	[K][L][V] = [0d60][0d2][0xAF D8]

US Key	x	ESD Digraph	Wl
US Name	x	ESD Name	Weapon Load

US Units	US Range	US Format	ESD Units	ESD Range	ESD Format
x	x	x	x	x	x

Notes	Notes
- x	- x

US Conversion	ESD Conversion
x	x
To US: - x	*To ESD:* - x
To LS: - x	*To LS:* - x

8.60.1 Example Weapon Load

Weapon load is broken into two bytes with the first byte indicates the aircraft store location, and the second byte indicates store type. Each byte is broken into two nibbles with [nib1] being the most significant nibble with bit order [3210] where 3=msb.

Aircraft store location is indicated by station number which starts numbering at the outboard left wing as store location 1 and increases towards the outboard right wing. Each station can have a different weapon installed, or multiple weapons on the same station. In a multiple weapon per station situation, the substation number begins at 1 and increases from there. A substation number of 0 indicates a single store located at the station. The Store Location byte has two nibbles with the first most significant nibble indicating station number, and the second indicating substation number. Note an example store location in the diagram of Figure 8-19:

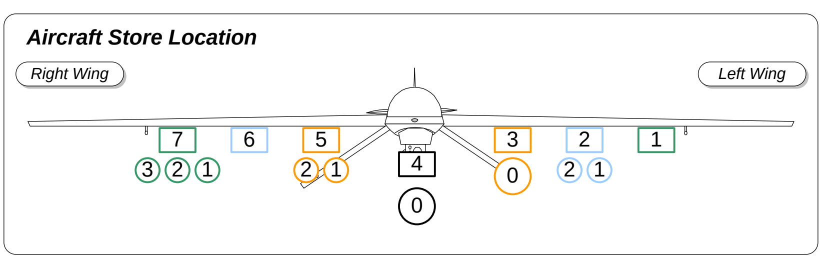

Figure 8-19: Aircraft Store Location

The weapon type byte is also broken into two nibbles with the first most significant nibble indicating weapon type and the second nibble indicating weapon variant.

A listing of available weapons is TBD.

8.61 Tag 61: Weapon Fired Conversion

		Units	Range	Format
LS Tag	61	uint8	x	nibble
LS Name	Weapon Fired			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 13 00 00 00			

Notes	Conversion Formula
- Indication when a particular weapon is released. Correlate with Unix Time stamp. - Identical format to Weapon Load byte 2: - [byteN] = [[nib1][nib2]] - nib1 = Station Number - nib2 = Substation Number	x x

Example Value	Example LS Packet
186	[K][L][V] = [0d61][0d2][0xBA]

US Key	x		ESD Digraph	Wf	
US Name	x		ESD Name	Weapon Fired	
Units	Range	Format	Units	Range	Format
x	x	x	x	x	x
Notes			Notes		
- x			- x		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			x *To ESD:* - x *To LS:* - x		

8.61.1 Example Weapon Fired

The Weapon Fired metadata item has the same format as the first byte of the Weapon Load metadata item indicating station and substation location of a store. Byte 1 is broken into two nibbles with [nib1] being the most significant nibble with bit order [3210] where 3=msb.

When included in a KLV packet, the weapon fired item should be correlated with the mandatory timestamp to determine the release time of a weapon.

8.62 Tag 62: Laser PRF Code Conversion

LS Tag	62	Units	Range	Format
LS Name	Laser PRF Code	None	0..65535	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 02 02 01 00 00 00			

Notes	Conversion Formula
- A laser's Pulse Repetition Frequency (PRF) code used to mark a target. - The Laser PRF code is a three or four digit number consisting of the values 1..8. - Only the values 1111..8888 can be used without 0's or 9's.	x x

Example Value	Example LS Packet
50895	[K][L][V] = [0d62][0d2][0xC6 CF]

US Key	x		ESD Digraph	Lc	
US Name	x		ESD Name	Laser PRF Code	
Units	Range	Format	Units	Range	Format
x	x	x	None	1111..8888	NNNN

Notes	Notes
- x	- A laser's Pulse Repetition Frequency (PRF) code used to mark a target. - The Laser PRF code is a three or four digit number consisting of the values 1..8. - Only the values 1111..8888 can be used without 0's or 9's.

US Conversion	ESD Conversion
x	x
To US: - x *To LS:* - x	*To ESD:* - Convert LS uint to ASCII. *To LS:* - Convert ASCII to LS uint.

8.62.1 Example Laser PRF Code

When enabled, laser designators can generate a pulsed signal according to a Pulse Repetition Frequency (PRF) Code which distinguishes one laser beam from another.

8.63 Tag 63: Sensor Field of View Name Conversion

LS Tag	63	Units	Range	Format
LS Name	Sensor Field of View Name	List	0..255	uint8
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 02 02 02 00 00 00			
Notes		Conversion Formula		
- Names sensor field of view quantized steps. - 00 = Ultranarrow - 01 = Narrow - 02 = Medium - 03 = Wide - 04 = Ultrawide - 05 = Narrow Medium - 06 = 2x Ultranarrow - 07 = 4x Ultranarrow		x x		
Example Value	Example LS Packet			
209	[K][L][V] = [0d63][0d1][0xD1]			

US Key	x		ESD Digraph	Vn	
US Name	x		ESD Name	Sensor Field of View Name	
Units	Range	Format	Units	Range	Format
x	x	x	Code	00..NN	NN
Notes			Notes		
- x			- Names sensor field of view quantized steps. - 00 = Ultranarrow - 01 = Narrow - 02 = Medium - 03 = Wide - 04 = Ultrawide - 05 = Narrow Medium - 06 = 2x Ultranarrow - 07 = 4x Ultranarrow		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			x *To ESD:* - Convert LS uint to ASCII. *To LS:* - Convert ASCII to LS uint.		

8.63.1 Example Sensor Field of View Name

The field of view name is a way to indicate to the operator the current lens used on the Motion Imagery sensor.

The Sensor Field of View names are for generic guidance and do not correspond to specific field of view values. Refer to Horizontal and Vertical Field of View metadata items (Tags 16 & 17) for specific aperture angles.

8.64 Tag 64: Platform Magnetic Heading Conversion

LS Tag	64	Units	Range	Format
LS Name	Platform Magnetic Heading	Degrees	0..360	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 08 00 00 00			
Notes		Conversion Formula		
- Aircraft magnetic heading angle. Relative between longitudinal axis and Magnetic North measured in the horizontal plane. - Map 0..(2^16-1) to 0..360. - Resolution: ~5.5 milli degrees.		$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right)$ $LS\_64\_dec = \left(\frac{360}{65535} * LS\_64\right)$		
Example Value		Example LS Packet		
311.8682 Degrees		[K][L][V] = [0d64][0d2][0xDD C5]		
US Key	x	ESD Digraph	Mh	
US Name	x	ESD Name	Platform Magnetic Heading	
Units / Range / Format	x / x / x	Units	Range	Format
		Degrees	0..359.99	DDD.HH
Notes		Notes		
- x		- Aircraft magnetic heading angle. Relative between fuselage chord line and Magnetic North.		
US Conversion		ESD Conversion		
x *To US:* - x *To LS:* - x		$ESD\_dec = \left(\frac{360}{65535} * LS\_uint\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ESD ASCII to decimal. - Map decimal to uint16.		

8.64.1 Example Magnetic Heading

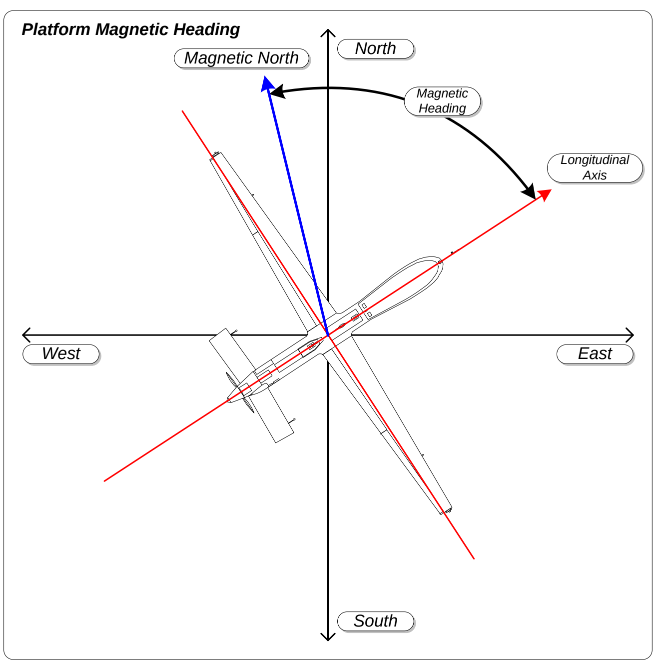

Figure 8-20: Magnetic Heading

8.65 Tag 65: UAS LS Version Number Conversion

LS Tag	65	Units	Range	Format
LS Name	UAS LS Version Number	Number	0..255	uint8
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 02 03 03 00 00 00			
Notes		Conversion Formula		
- Version number of the UAS LS document used to generate a source of UAS LS KLV metadata. - 0 is pre-release, initial release (0601.0), or test data. - 1..255 corresponds to document revisions ST0601.1 thru ST0601.255.		x x		
Example Value	Example LS Packet			
Version 232	[K][L][V] = [0d65][0d1][0xE8]			

US Key	x		ESD Digraph	Iv	
US Name	x		ESD Name	ESD ICD Version	
Units	Range	Format	Units	Range	Format
x	x	x	Number	0..99	NN
Notes			Notes		
- x			- Version of the ESD System use to encode ESD Data. - 0 corresponds to documents ASI-119 and ASI-209. - 1..99 corresponds to document revisions of ST0601.1 thru ST 0601.99.		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			x *To ESD:* - Convert uint to ASCII. *To LS:* - Convert ASCII to uint.		

8.65.1 Example UAS LS Version Number

The UAS LS version number metadata item is used to indicate which version of ST 0601 is used as the source standard of UAS LS metadata. This item is not required in every packet of metadata, but is useful when included periodically.

8.66 Tag 66: Target Location Covariance Matrix Conversion

		Units	Range	Format
LS Tag	66	TBD	TBD	TBD
LS Name	Target Location Covariance Matrix			
US Mapped Key	06 0E 2B 34 02 05 01 01 0E 01 03 03 14 00 00 00			
Notes		**Conversion Formula**		
- Covariance Matrix of the error associated with a targeted location. - Details TBD.		TBD TBD		
Example Value	**Example LS Packet**			
x	[K][L][V] = [0d66][0dTBD][x]			

8.66.1 Example Target Location Covariance Matrix

Details TBD

8.67 Tag 67: Alternate Platform Latitude Conversion

		Units	Range	Format
LS Tag	67	Degrees	+/- 90	int32
LS Name	Alternate Platform Latitude			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 14 00 00 00			

Notes	Conversion Formula
- Alternate Platform Latitude. Represents latitude of platform connected with UAS. - Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-90. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~42 nano degrees.	$LS\_dec = \left(\frac{LS\ range}{int\_range} * LS\_int\right)$ $LS\_67\_dec = \left(\frac{180}{4294967294} * LS\_67\right)$

Example Value	Example LS Packet
-86.041207348947 Degrees	[K][L][V] = [0d67][0d4][0x85 A1 5A 39]

8.67.1 Example Latitude

Latitude is the angular distance north or south of the earth's equator, measured in degrees along a meridian. Generated from GPS/INS information and based on the WGS84 coordinate system.

The Alternate Platform is an airborne or ground based platform that is connected via direct datalink to a UAS generating Motion Imagery and metadata.

Note that the int32 used in the LS value is encoded using two's complement.

8.68 Tag 68: Alternate Platform Longitude Conversion

		Units	Range	Format
LS Tag	68	Degrees	+/- 180	int32
LS Name	Alternate Platform Longitude			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 15 00 00 00			

Notes	Conversion Formula
- Alternate Platform Longitude. Represents longitude of platform connected with UAS. - Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-180. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~84 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_68\_dec = \left(\frac{360}{4294967294} * LS\_68\right)$

Example Value	Example LS Packet
0.155527554524842 Degrees	[K][L][V] = [0d68][0d4][0x00 1C 50 1C]

8.68.1 Example Longitude

Longitude is the angular distance on the earth's surface, measured east or west from the prime meridian at Greenwich, England, to the meridian passing through a position of interest. Generated from GPS/INS information and based on the WGS84 coordinate system.

The Alternate Platform is an airborne or ground based platform that is connected via direct datalink to a UAS generating Motion Imagery and metadata.

Note that the int32 used in the LS value is encoded using two’s complement.

8.69 Tag 69: Alternate Platform Altitude Conversion

LS Tag	69	Units	Range	Format
LS Name	Alternate Platform Altitude	Meters	-900..19000	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 16 00 00 00			
Notes		Conversion Formula		
- Altitude of alternate platform as measured from Mean Sea Level (MSL). Represents altitude of platform connec ted with UAS. - Map 0..(2^16-1) to -900..19000 meters. - 1 meter = 3.2808399 feet. - Resolution: ~0.3 meters.		$LS\_dec = \left(\frac{LS\_range}{uint\_range} * LS\_uint\right) - Offset$ $LS\_69\_dec = \left(\frac{19900}{65535} * LS\_69\right) - 900$		
Example Value	Example LS Packet			
9.445334 Meters	[K][L][V] = [0d69][0d2][0x0B B3]			

8.69.1 Example Platform Altitude

The Alternate Platform Altitude is a true altitude or true vertical distance above mean sea level. Measurement is GPS derived.

The Alternate Platform is an airborne or ground based platform that is connected via direct datalink to a UAS generating Motion Imagery and metadata.

8.70 Tag 70: Alternate Platform Name Conversion

LS Tag	70	Units	Range	Format
LS Name	Alternate Platform Name	String	1..127	ISO 646
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 17 00 00 00			
Notes		Conversion Formula		
- Name of alternate platform connected to UAS. - E.g.: 'Apachce', 'Rover', 'Predator', 'Reaper', 'Outrider', 'Pioneer', 'IgnatER', 'Warrior', 'Shadow', 'Hunter II', 'Global Hawk', 'Scan Eagle', etc. - Value field is Free Text. - Maximum 127 characters.		x x		
Example Value	Example LS Packet			
APACHE	[K][L][V] = [0d70][0d6][0x41 50 41 43 48 45]			

8.70.1 Example Alternate Platform Name

The Alternate Platform Name metadata item distinguishes which platform is connected with the UAS which is generating Motion Imagery and metadata products. The alternate platform can be airborne or ground based and is to be described sufficiently (yet with brevity) in text using this metadata item.

The Alternate Platform is an airborne or ground based platform that is connected via direct datalink to a UAS generating Motion Imagery and metadata.

8.71 Tag 71: Alternate Platform Heading Conversion

LS Tag	71	Units	Range	Format
LS Name	Alternate Platform Heading	Degrees	0..360	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 01 18 00 00 00			
Notes		Conversion Formula		
- Heading angle of alternate platform connected to UAS. Relative between longitudinal axis and True North measured in the horizontal plane. - Map 0..(2^16-1) to 0..360. - Resolution: ~5.5 milli degrees.		$LS\_dec = \left(\frac{LS\ range}{int\_range} * LS\_int\right)$ $LS\_71\_dec = \left(\frac{360}{65535} * LS\_71\right)$		
Example Value	Example LS Packet			
32.60242 Degrees	[K][L][V] = [0d71][0d2][0x17 2F]			

8.71.1 Example Alternate Platform Heading

The Alternate Platform heading angle is defined as the angle between the alternate platform longitudinal axis (line made by the fuselage) and true north measured in the horizontal plane. Angles increase in a clockwise direction when looking from above the platform. North is 0 degrees, east is 90, south is 180, and west is 270 degrees from true north.

The Alternate Platform is an airborne or ground based platform that is connected via direct datalink to a UAS generating Motion Imagery and metadata.

8.72 Tag 72: Event Start Time - UTC Conversion

LS Tag	72	Units	Range	Format
LS Name	Event Start Time - UTC	Microseconds	0..(2^64-1)	uint64
US Mapped Key	Use EG0104 US Key			
Notes		Conversion Formula		
- Start time of scene, project, event, mission, editing event, license, publication, etc. - Represented as the microseconds elapsed since midnight (00:00:00), January 1, 1970. - Derived from the POSIX IEEE 1003.1 standard. - Resolution: 1 microsecond.		x x		
Example Value		Example LS Packet		
April 16, 1995. 13:44:54		[K][L][V] = [0d72][0d8][0x00 02 D5 CF 4D DC 9A 35]		

US Key	06 0E 2B 34 01 01 01 01 07 02 01 02 07 01 00 00		ESD Digraph	x	
US Name	Event Start Date Time - UTC		ESD Name	Mission Start Time, Date, and Date of Collection	
Units	Range	Format	Units	Range	Format
Date/Time	'YYYYMMDDhhmmss'	ISO 8601	x	x	x
Notes			Notes		
- The absolute beginning date and time of the project, mission, scene, editing event, license, publication etc. - Formatted text as: 'YYYYMMDDhhmmss'			- The LS Event Start Time - UTC can be converted to three ESD items: - Mission Start Date (Md) - Mission Start Time (Mc) - Date of Collection (Cd) - Refer to EG0104 for details on these ESD items.		
US Conversion			ESD Conversion		
x *To US:* - Convert uint64 to formatted string. *To LS:* - Convert formatted string to uint64.			x *To ESD:* - x *To LS:* - x		

8.72.1 Example Event Start Time – UTC

A Precision Time Stamp discretely labels a scale of time. This system is widely used within systems of differing underlying architectures. The Precision Time Stamp is an encoding of Coordinated Universal Time (UTC) and therefore accounts for the addition (or subtraction) of leap seconds. Leap seconds are used to synchronize the UTC clock metric with the yearly rotation period of the earth about the sun.

This POSIX time metadata value is used to represent the start time of a mission, or other event related to the Motion Imagery collection.

Event Start Time is to be interpreted as an arbitrary time hack indicating the start of some event.

8.73 Tag 73: RVT Local Set Conversion

LS Tag LS Name US Mapped Key	73 RVT Local Set Use ST0806 RVT LS 16-byte Key.	Units	Range	Format
		None	Set	Set
Notes		Conversion Formula		
- Local set tag to include the ST0806 RVT Local Set metadata items within ST0601. Use the ST0806 Local Set Tags within the ST0601 tag 0d73. - The length field is the size of all RVT LS metadata items to be packaged within tag 0d73.		x x		
Example Value		Example LS Packet		
x		[K][L][V] = [0d73][0dx][x]		

US Key	06 0E 2B 34 02 0B 01 01 0E 01 03 01 02 00 00 00		ESD Digraph	x	
US Name	Remote Video Terminal Local Set		ESD Name	x	
Units	Range	Format	Units	Range	Format
None	Set	Set	x	x	x
Notes			Notes		
- x			- x		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			x *To ESD:* - x *To LS:* - x		

8.73.1 Example RVT Local set Conversion

ST 0601 Tag 73 allows users to include, or nest, RVT LS (ST 0806) metadata items within ST 0601.

This provides users who are required to use the RVT LS data field (Points of Interest, Areas of Interest, etc.) a method to leverage the data field contained within ST 0601 (like platform location, and sensor pointing angles).

8.74 Tag 74: VMTI Data Set Conversion

LS Tag	74	Units	Range	Format
LS Name	VMTI Data Set	None	Set	Set
US Mapped Key	Use ST0903 VMTI LS 16-byte Key.			
Notes		Conversion Formula		
- Local set tag to include the ST0903 VMTI Local Set metadata items within ST0601. Use the ST0903 Local Set Tags within the ST0601 tag 0d74. - The length field is the size of all VMTI LS metadata items to be packaged within tag 0d74.		x x		
Example Value		Example LS Packet		
x		[K][L][V] = [0d74][0dx][x]		

US Key	06 0E 2B 34 02 0B 01 01 0E 01 03 03 06 00 00 00		ESD Digraph	x	
US Name	Video Moving Target Indicator Local Set		ESD Name	x	
Units	Range	Format	Units	Range	Format
None	Set	Set	x	x	x
Notes			Notes		
- x			- x		
US Conversion			ESD Conversion		
x *To US:* - x *To LS:* - x			x *To ESD:* - x *To LS:* - x		

8.74.1 Example VMTI Local set Conversion

ST 0601 Tag 74 allows users to include, or nest, VMTI LS (MISB ST 0903) metadata items within ST 0601.

This provides users who are required to use the VMTI LS data field a method to leverage the data field contained within ST 0601 (like platform location, and sensor pointing angles, or frame center).

8.75 Tag 75: Sensor Ellipsoid Height Conversion

LS Tag	75	**Units**	**Range**	**Format**
LS Name	Sensor Ellipsoid Height	Meters	-900..19000	uint16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 02 01 82 47 00 00			

Notes	Conversion Formula
- Sensor Ellipsoid Height as measured from the reference WGS84 Ellipsoid. - Map 0..(2^16-1) to -900..19000 meters. - 1 meter = 3.2808399 feet. - Resolution: ~0.3 meters.	$LS\_dec = \left(\frac{LS\ range}{uint\_range} * LS\_uint\right) - Offset$ $LS\_75\_dec = \left(\frac{19900}{65535} * LS\_75\right) - 900$

Example Value	Example LS Packet
14190.72 Meters	[K][L][V] = [0d75][0d2][0xC2 21]

8.75.1 Example Sensor Ellipsoid Height

For legacy purposes, both MSL (Tag 15) and HAE (Tag 75) representations of Sensor True Altitude MAY appear in the same ST 0601 packet. A single representation is preferred favoring the HAE version (Tag 75).

The Sensor Ellipsoid Height is the vertical distance between the sensor and the WGS84 Reference Ellipsoid. Measurement is GPS derived.

8.76 Tag 76: Alternate Platform Ellipsoid Height Conversion

		Units	Range	Format
LS Tag	76	Meters	-900..19000	uint16
LS Name	Alternate Platform Ellipsoid Height			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 02 01 82 48 00 00			

Notes	Conversion Formula
- Alternate Platform Ellipsoid Height as measured from the reference WGS84 Ellipsoid. - Map 0..(2^16-1) to -900..19000 meters. - 1 meter = 3.2808399 feet. - Resolution: ~0.3 meters.	$LS\_dec = \left(\frac{LS\ range}{uint\_range} * LS\_uint\right) - Offset$ $LS\_76\_dec = \left(\frac{19900}{65535} * LS\_76\right) - 900$

Example Value	Example LS Packet
9.445334 Meters	[K][L][V] = [0d76][0d2][0x0B B3]

8.76.1 Example Alternate Platform Ellipsoid Height

The Alternate Platform Ellipsoid Height is the vertical distance between the sensor and the WGS84 Reference Ellipsoid. Measurement is GPS derived.

The Alternate Platform is an airborne or ground based platform that is connected via direct datalink to a UAS generating Motion Imagery and metadata.

8.77 Tag 77: Operational Mode Conversion

LS Tag	77	Units	Range	Format
LS Name	Operational Mode	None	None	uint8
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 01 03 21 00 00 00			
Notes		Conversion Formula		
- Indicates the mode of operations of the event portrayed in metadata. Enumerated. - 0x00 = "Other" - 0x01 = "Operational" - 0x02 = "Training" - 0x03 = "Exercise" - 0x04 = "Maintenance" - 0x05 = "Test"		x x		
Example Value	Example LS Packet			
x	[K][L][V] = [0d77][0dx][x]			

8.77.1 Example Operational Mode

The Operational Mode provides an indication of the event portrayed in the metadata. This allows for categorization of Motion Imagery streams and is often useful for archival systems.

8.78 Tag 78: Frame Center Height Above Ellipsoid Conversion

		Units	Range	Format
LS Tag	78	Meters	-900..19000	uint16
LS Name	Frame Center Height Above Ellipsoid			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 02 03 48 00 00 00			

Notes	Conversion Formula
- Frame Center Ellipsoid Height as measured from the reference WGS84 Ellipsoid. - Map 0..(2^16-1) to -900..19000 meters. - 1 meter = 3.2808399 feet. - Resolution: ~0.3 meters.	$LS\_dec = \left(\frac{LS\ range}{uint\_range} * LS\_uint\right) - Offset$ $LS\_78\_dec = \left(\frac{19900}{65535} * LS\_78\right) - 900$

Example Value	Example LS Packet
9.445334 Meters	[K][L][V] = [0d78][0d2][0x0B B3]

For legacy purposes, both MSL (Tag 25) and HAE (Tag 78) representations of Frame Center Elevation MAY appear in the same ST 0601 packet. A single representation is preferred favoring the HAE version (Tag 78).

The Frame Center Ellipsoid Height is the vertical distance on the ground within the center of the Motion Imagery frame and the WGS84 Reference Ellipsoid. Measurement is GPS derived.

8.79 Tag 79: Sensor North Velocity Conversion

LS Tag	79	Units	Range	Format
LS Name	Sensor North Velocity	Meters/Sec	+/-327	int16
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 02 02 7E 00 00 00			
Notes		**Conversion Formula**		
- Northing velocity of the sensor or platform. Positive towards True North - Map-(2^15-1)..(2^15-1) to +/-327 - Use -(2^15) as an "out of range" indicator. - -(2^15) = 0x8000. - Resolution: ~ 1 cm/sec.		$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_79 = \left(\frac{654}{65534} * LS\_79\right)$		
Example Value	**Example LS Packet**			
x	[K][L][V] = [0d79][0dx][x]			

8.79.1 Example Sensor North Velocity

The Northing velocity of the sensor is the sensor movement rate in the north direction. Positive values indicate a sensor approaching True North.

Note that the int16 used in the LS value is encoded using two's complement.

8.80 Tag 80: Sensor East Velocity Conversion

		Units	Range	Format
LS Tag	80	Meters/Sec	+/-327	int16
LS Name	Sensor East Velocity			
US Mapped Key	06 0E 2B 34 01 01 01 01 0E 01 02 02 7F 00 00 00			

Notes	Conversion Formula
- Easting velocity of the sensor or platform. Positive towards East. - Map-(2^15-1)..(2^15-1) to +/-327 - Use -(2^15) as an "out of range" indicator. - -(2^15) = 0x8000. - Resolution: ~ 1 cm/sec.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_80 = \left(\frac{654}{65534} * LS\_80\right)$

Example Value	Example LS Packet
x	[K][L][V] = [0d80][0dx][x]

8.80.1 Example Sensor East Velocity

The Easting velocity of the sensor is the sensor movement rate in the east direction. Positive values indicate a sensor approaching east.

Note that the int16 used in the LS value is encoded using two's complement.

8.81 Tag 81: Image Horizon Pixel Pack Conversion

		Units	Range	Format
LS Tag	81	Pack	Pack	Pack
LS Name	Image Horizon Pixel Pack			
US Mapped Key	06 0E 2B 34 02 05 01 01 0E 01 03 02 08 00 00 00			
Notes		**Conversion Formula**		
- <tag 81><length> - < start x0, start y0 // point p0 - end x1, end y1 // point p1 - start lat, start lon - end lat, end lon - >		See Notes below.		
Example Value	**Example LS Packet**			
x	[K][L][V] = [0d81][0dx][x]			

8.81.1 Description of Image Horizon Pixel Pack

The Image Horizon Pixel Pack allows a user to separate sky and ground portions of an image by defining a line representing the horizon. The method for detecting where the horizon is within the image is left to the system implementer.

The line representing the horizon which transects the image is defined by a vector with start and end points which must lie on the extents of the image. This is called the Horizon Vector. The horizontal (x) and vertical (y) coordinates are represented in a relative scale (from 0 to 100%) with (x,y) equal to (0%,0%) being the top left corner of the image.

Once start and end coordinates are defined, the pixels to the right of this Horizon Vector designates the ground region, while pixels to the left represent sky. Refer to Figure 8-21.

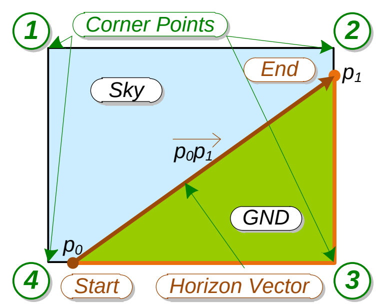

Figure 8-21: Horizon Vector

With the Horizon Vector defined, only the image corner points to the right are considered valid and allowed to be included within a ST 0601 packet. No invalid corner coordinates are allowed when the Image Horizon Pixel Pack is included in the same ST 0601 packet.

The Horizon Line and valid corner coordinates define the Pixel Frame (PF) (i.e. a polygon) which represents ground pixels.

In the example shown in in Figure 8-21, corner point number 3 is the only valid corner point and is used with the start and end points to define a 3-point Pixel Frame.

Examples for 3-point, 4-point, and 5-point Pixel Frames are shown in Figure 8-22.

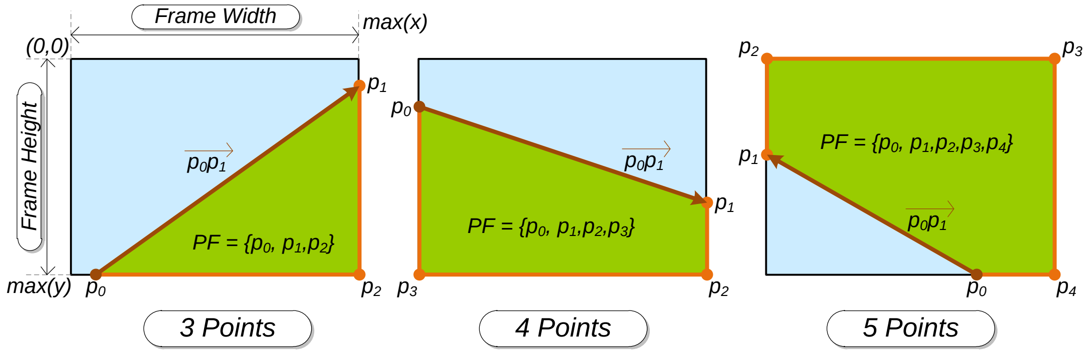

Figure 8-22: Pixel Frame Examples

Note that the pixel points p_0 through p_4 do not always directly correspond with the offset (Tags 26-33) or absolute (Tags 82-89) corner coordinates defined within this document.

8.81.2 Image Horizon Pixel Pack Example

To show how to use the Image Horizon Pixel Pack, consider the following example shown in Figure 8-23 for sample 720p airborne imagery:

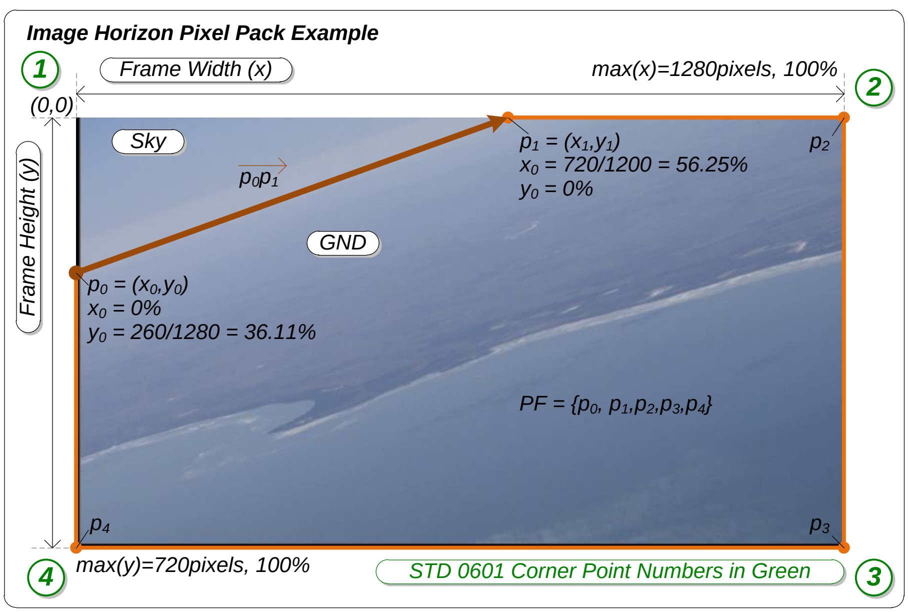

Figure 8-23: Image Horizon Pixel Pack Example

In the example above, the horizon (barely visible through haze) is covered by the Horizon Vector with p_0 = (0%, 36.11%), and p_1 = (56.25%, 0).

8.81.3 Decoding the Image Horizon Pixel Pack

When an Image Horizon Pixel Pack only includes the x & y coordinates of the Horizon Vector and not the geo-locations, the Horizon Vector is used to determine the image pixel coordinates (derived from the relative values) which construct the Pixel Frame.

When the latitudes and longitudes of the Horizon Vector are included, these geo-locations along with the valid offset or absolute corner coordinates in the same ST 0601 packet are then matched with the appropriate points defined by the Pixel Frame.

8.81.4 Floating Length Pack Definition for the Image Horizon Pixel Pack

The Image Horizon Pixel Pack makes use of a Floating Length Pack as described by MISB RP 0701 and allows a user to include or exclude data items as necessary. The first items defined within this pack are the start and end x & y coordinates representing the start and end of the Horizon Vector. These are then followed by real earth latitude-longitude geo-coordinate pairs for the start and end points of the Horizon Vector.

As used here, the minimum required components are the start and end x & y points defining the Horizon Vector in image space, and the latitudes/longitudes of these points are optional (but recommended). Contents are defined in Table 2:

Table 2: Image Horizon Pixel Pack

Local Set Key	Name
06 0E 2B 34 - 02 05 01 01 - 0E 01 03 02 - 08 00 00 00 (CRC 37658)	Image Horizon Pixel Pack

Constituent Elements					
Key	**Name**	**Notes**	**Units/Range**	**Format**	**Len**
06 0E 2B 34 01 01 01 01 0E 01 01 02 09 01 00 00 (CRC 3334)	Start x0	The X coordinate (in percent) of an X-Y pair representing the start point of a vector crossing an image. Top left of image is 0,0 with positive X increasing to the right. To be used with Start y0. Mandatory in the Image Horizon Pixel Pack.	Percent [0..100]	Uint8	1
06 0E 2B 34 01 01 01 01 0E 01 01 02 09 02 00 00 (CRC 21590)	Start y0	The Y coordinate (in percent) of an X-Y pair representing the start point of a vector crossing an image. Top left of image is 0.0 with positive Y increasing down. To be used with Start x0. Mandatory in the Image Horizon Pixel Pack.	Percent [0..100]	Uint8	1
06 0E 2B 34 01 01 01 01 0E 01 01 02 09 03 00 00 (CRC 25446)	End x1	The X coordinate (in percent) of an X-Y pair representing the end point of a vector crossing an image. Top left of image is 0,0 with positive X increasing to the right. To be used with End y0. Mandatory in the Image Horizon Pixel Pack.	Percent [0..100]	Uint8	1
06 0E 2B 34 01 01 01 01 0E 01 01 02 09 04 00 0 (CRC 59126)	End y1	The Y coordinate (in percent) of an X-Y pair representing the end point of a vector crossing an image. Top left of image is 0.0 with positive Y increasing down. To be used with End x0. Mandatory in the Image Horizon Pixel Pack.	Percent [0..100]	Uint8	1
06 0E 2B 34 01 01 01 01 0E 01 01 02 09 05 00 00 (CRC 53702)	Start Latitude	The Latitude of the Start point (x0,y0) on the image border. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use (-2^31) as an “error” indicator. Optional (but recommended).	Degrees [-90..+90]	Int32	4
06 0E 2B 34 01 01 01 01 0E 01 01 02 09 06 00 00 (CRC 34966)	Start Longitude	The Longitude of the Start point (x0,y0) on the image boarder. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use (-2^31) as an “error” indicator. Optional (but recommended).	Degrees [-180..+180]	Int32	4
06 0E 2B 34 01 01 01 01 0E 01 01 02 09 07 00 00 (CRC 49062)	End Latitude	The Latitude of the End point (x1,y1) on the image boarder. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-90. Use (-2^31) as an “error” indicator. Optional (but recommended).	Degrees [-90..+90]	Int32	4
06 0E 2B 34 01 01 01 01 0E 01 01 02 09 08 00 00 (CRC 37783)	End Longitude	The Longitude of the End point (x1,y1) on the image boarder. Based on WGS84 ellipsoid. Map -(2^31-1)..(2^31-1) to +/-180. Use (-2^31) as an “error” indicator. Optional (but recommended).	Degrees [-180..+180]	Int32	4

8.82 Tag 82: Corner Latitude Point 1 (Full) Conversion

LS Tag	82	Units	Range	Format
LS Name	Corner Latitude Point 1 (Full)	Degrees	+/- 90	int32
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Latitude for upper left corner. - Full Range. - Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-90. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~42 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_82\_dec = \left(\frac{180}{4294967294} * LS\_82\right)$

Example Value	Example LS Packet
-10.579637999887 Corrected Degrees	[K][L][V] = [0d82][0d2][0xC0 6E]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 07 01 00	ESD Digraph	Rg
US Name	Corner Latitude Point 1 (Decimal Degrees)	ESD Name	SAR Latitude 4

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes	Notes
- Latitude coordinate of corner 1 of an image or bounding rectangle. - Positive (+) is northern hemisphere. - Negative (-) is southern hemisphere.	- The latitude of the upper left corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$ *To US:* - US = (double)(180/0xFFFFFFFE * LS) *To LS:* - LS = (int32)round(0xFFFFFFFE/180 * US)	$ESD\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int32.

8.82.1 Example Corner Latitude Point 1 (Full) Conversion

For legacy purposes, both range-restricted (Tags 26-33) and full-range (Tag 82-89) representations of Image Corner Coordinates MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tags 82-89) being favored as per Section 6.3.

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, should either not be reported, or be reported as an “error”.

Corner point 1 is the upper left corner of the captured image as highlighted in red (Figure 8-24).

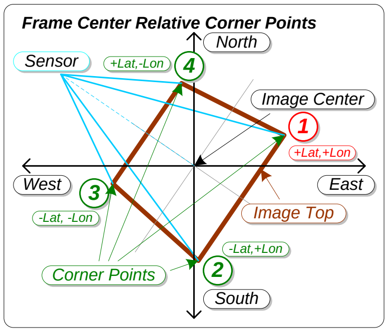

Figure 8-24: Offset Corner Point 1

Value is encoded using two’s complement.

8.83 Tag 83: Corner Longitude Point 1 (Full) Conversion

LS Tag	83	Units	Range	Format
LS Name	Corner Longitude Point 1 (Full)	Degrees	+/- 180	int32
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Longitude for upper left corner. - Full Range. - Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-180. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~84 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_83\_dec = \left(\frac{360}{4294967294} * LS\_83\right)$

Example Value	Example LS Packet
29.1273677986333 Corrected Degrees	[K][L][V] = [0d83][0d2][0xCB E9]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0B 01 00		ESD Digraph	Rh	
US Name	Corner Longitude Point 1 (Decimal Degrees)		ESD Name	SAR Longitude 4	
Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes	Notes
- Longitude coordinate of corner 1 of an image or bounding rectangle. - Positive (+) is eastern hemisphere. - Negative (-) is western hemisphere.	- The longitude of the upper left corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$ *To US:* - US = (double)(360/0xFFFFFFFE * LS) *To LS:* - LS = (int32)round(0xFFFFFFFE/360 * US)	$ESD\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int32.

8.83.1 Example Corner Longitude Point 1 (Full) Conversion

For legacy purposes, both range-restricted (Tags 26-33) and full-range (Tag 82-89) representations of Image Corner Coordinates MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tags 82-89) being favored as per Section 6.3.

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth (an example being the tracking of an airborne object), should either not be reported, or be reported as an "error".

Corner point 1 is the upper left corner of the captured image. See Figure 8-24 for Tag 82 above.

8.84 Tag 84: Corner Latitude Point 2 (Full) Conversion

LS Tag / LS Name / US Mapped Key		Units	Range	Format
LS Tag	84	Degrees	+/- 90	int32
LS Name	Corner Latitude Point 2 (Full)			
US Mapped Key	Use EG0104 US Key			

Notes

- Frame Latitude for upper right corner.
- Full Range.
- Based on WGS84 ellipsoid.
- Map -(2^31-1)..(2^31-1) to +/-90.
- Use -(2^31) as an "error" indicator.
- -(2^31) = 0x80000000.
- Resolution: ~42 nano degrees.

Conversion Formula

$$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$$

$$LS\_84\_dec = \left(\frac{180}{4294967294} * LS\_84\right)$$

Example Value	Example LS Packet
-10.5661816260963 Corrected Degrees	[K][L][V] = [0d84][0d2][0xD7 65]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 08 01 00	ESD Digraph	Ra
US Name	Corner Latitude Point 2 (Decimal Degrees)	ESD Name	SAR Latitude 1

US Units	US Range	US Format	ESD Units	ESD Range	ESD Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes (US)

- Latitude coordinate of corner 2 of an image or bounding rectangle.
- Positive (+) is northern hemisphere.
- Negative (-) is southern hemisphere.

Notes (ESD)

- The latitude of the upper right corner of the SAR image box.

US Conversion

$$US\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$$

To US:
- US = (double)(180/0xFFFFFFFE * LS)

To LS:
- LS = (int32)round(0xFFFFFFFE/180 * US)

ESD Conversion

$$ESD\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$$

To ESD:
- Convert LS to decimal.
- Convert decimal to ASCII.

To LS:
- Convert ASCII to decimal.
- Map decimal to int32.

8.84.1 Example Corner Latitude Point 2 (Full) Conversion

For legacy purposes, both range-restricted (Tags 26-33) and full-range (Tag 82-89) representations of Image Corner Coordinates MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tags 82-89) being favored as per Section 6.3.

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, should either not be reported, or be reported as an “error”.

Corner point 2 is the upper right corner of the captured image as highlighted in red (Figure 8-25).

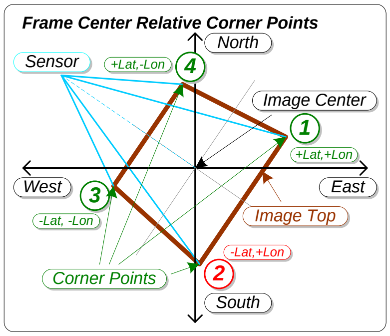

Figure 8-25: Offset Corner Point 2

Value is encoded using two’s complement.

8.85 Tag 85: Corner Longitude Point 2 (Full) Conversion

LS Tag	85	Units	Range	Format
LS Name	Corner Longitude Point 2 (Full)	Degrees	+/- 180	int32
US Mapped Key	Use EG0104 US Key			

Notes
- Frame Longitude for upper right corner.
- Full Range.
- Based on WGS84 ellipsoid.
- Map -(2^31-1)..(2^31-1) to +/-180.
- Use -(2^31) as an "error" indicator.
- -(2^31) = 0x80000000.
- Resolution: ~84 nano degrees.

Conversion Formula

$$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$$

$$LS\_85\_dec = \left(\frac{360}{4294967294} * LS\_85\right)$$

Example Value	Example LS Packet
29.140824172424 Corrected Degrees	[K][L][V] = [0d85][0d2][0xE2 E0]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0C 01 00	ESD Digraph	Rb
US Name	Corner Longitude Point 2 (Decimal Degrees)	ESD Name	SAR Longitude 1

Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes (US)
- Longitude coordinate of corner 2 of an image or bounding rectangle.
- Positive (+) is eastern hemisphere.
- Negative (-) is western hemisphere.

Notes (ESD)
- The longitude of the upper right corner of the SAR image box.

US Conversion

$$US\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$$

To US:
- US = (double)(360/0xFFFFFFFE * LS)

To LS:
- LS = (int32)round(0xFFFFFFFE/360 * US)

ESD Conversion

$$ESD\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$$

To ESD:
- Convert LS to decimal.
- Convert decimal to ASCII.

To LS:
- Convert ASCII to decimal.
- Map decimal to int32.

8.85.1 Example Corner Longitude Point 2 (Full) Conversion

For legacy purposes, both range-restricted (Tags 26-33) and full-range (Tag 82-89) representations of Image Corner Coordinates MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tags 82-89) being favored as per Section 6.3.

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth (an example being the tracking of an airborne object), should either not be reported, or be reported as an “error”.

Corner point 2 is the upper right corner of the captured image. See Figure 8-25 for Tag 84 above.

8.86 Tag 86: Corner Latitude Point 3 (Full) Conversion

		Units	Range	Format
LS Tag	86	Degrees	+/- 90	int32
LS Name	Corner Latitude Point 3 (Full)			
US Mapped Key	Use EG0104 US Key			

Notes
- Frame Latitude for lower right corner.
- Full Range.
- Based on WGS84 ellipsoid.
- Map -(2^31-1)..(2^31-1) to +/-90.
- Use -(2^31) as an "error" indicator.
- -(2^31) = 0x80000000.
- Resolution: ~42 nano degrees.

Conversion Formula

$$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$$

$$LS\_86\_dec = \left(\frac{180}{4294967294} * LS\_86\right)$$

Example Value	Example LS Packet
-10.5527275411938 Corrected Degrees	[K][L][V] = [0d86][0d2][0xEE 5B]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 09 01 00	ESD Digraph	Rc
US Name	Corner Latitude Point 3 (Decimal Degrees)	ESD Name	SAR Latitude 2

US Units	US Range	US Format	ESD Units	ESD Range	ESD Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes (US)
- Latitude coordinate of corner 3 of an image or bounding rectangle.
- Positive (+) is northern hemisphere.
- Negative (-) is southern hemisphere.

Notes (ESD)
- The latitude of the lower right corner of the SAR image box.

US Conversion

$$US\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$$

To US:
- US = (double)(180/0xFFFFFFFE * LS)

To LS:
- LS = (int32)round(0xFFFFFFFE/180 * US)

ESD Conversion

$$ESD\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$$

To ESD:
- Convert LS to decimal.
- Convert decimal to ASCII.

To LS:
- Convert ASCII to decimal.
- Map decimal to int32.

8.86.1 Example Corner Latitude Point 3 (Full) Conversion

For legacy purposes, both range-restricted (Tags 26-33) and full-range (Tag 82-89) representations of Image Corner Coordinates MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tags 82-89) being favored as per Section 6.3.

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, should either not be reported, or be reported as an “error”.

Corner point 3 is the lower right corner of the captured image as highlighted in red (Figure 8-26).

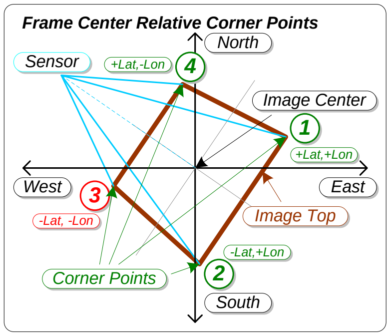

Figure 8-26: Offset Corner Point 3

Value is encoded using two's complement.

8.87 Tag 87: Corner Longitude Point 3 (Full) Conversion

		Units	Range	Format
LS Tag	87	Degrees	+/- 180	int32
LS Name	Corner Longitude Point 3 (Full)			
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Frame Longitude for lower right corner. - Full Range. - Based on WGS84 ellipsoid. - Map -(2^31-1)..(2^31-1) to +/-180. - Use -(2^31) as an "error" indicator. - -(2^31) = 0x80000000. - Resolution: ~84 nano degrees.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_87\_dec = \left(\frac{360}{4294967294} * LS\_87\right)$

Example Value	Example LS Packet
29.1542782573265 Corrected Degrees	[K][L][V] = [0d87][0d2][0xF9 D6]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0D 01 00	ESD Digraph	Rd
US Name	Corner Longitude Point 3 (Decimal Degrees)	ESD Name	SAR Longitude 2

Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes	Notes
- Longitude coordinate of corner 3 of an image or bounding rectangle. - Positive (+) is eastern hemisphere. - Negative (-) is western hemisphere.	- The longitude of the lower right corner of the SAR image box.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$ *To US:* - US = (double)(360/0xFFFFFFFE * LS) *To LS:* - LS = (int32)round(0xFFFFFFFE/360 * US)	$ESD\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$ *To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int32.

8.87.1 Example Corner Longitude Point 3 (Full) Conversion

For legacy purposes, both range-restricted (Tags 26-33) and full-range (Tag 82-89) representations of Image Corner Coordinates MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tags 82-89) being favored as per Section 6.3.

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth (an example being the tracking of an airborne object), should either not be reported, or be reported as an “error”.

Corner point 3 is the lower right corner of the captured image. See Figure 8-26 for Tag 86 above.

8.88 Tag 88: Corner Latitude Point 4 (Full) Conversion

LS Tag	88	Units	Range	Format
LS Name	Corner Latitude Point 4 (Full)	Degrees	+/- 90	int32
US Mapped Key	Use EG0104 US Key			

Notes
- Frame Latitude for lower left corner.
- Full Range.
- Based on WGS84 ellipsoid.
- Map -(2^31-1)..(2^31-1) to +/-90.
- Use -(2^31) as an "error" indicator.
- -(2^31) = 0x80000000.
- Resolution: ~42 nano degrees.

Conversion Formula

US = (double)(180/0xFFFFFFFE * LS)

$$LS\_88\_dec = \left(\frac{180}{4294967294} * LS\_88\right)$$

Example Value	Example LS Packet
-10.5392711674031 Corrected Degrees	[K][L][V] = [0d88][0d2][0x05 52]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0A 01 00	ESD Digraph	Re
US Name	Corner Latitude Point 4 (Decimal Degrees)	ESD Name	SAR Latitude 3

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Double	Degrees	+/- 90.00	PDDMMSST

Notes (US)
- Latitude coordinate of corner 4 of an image or bounding rectangle.
- Positive (+) is northern hemisphere.
- Negative (-) is southern hemisphere.

Notes (ESD)
- The latitude of the lower left corner of the SAR image box.

US Conversion

$$US\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$$

To US:
- US = (double)(180/0xFFFFFFFE * LS)

To LS:
- LS = (int32)round(0xFFFFFFFE/180 * US)

ESD Conversion

$$ESD\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$$

To ESD:
- Convert LS to decimal.
- Convert decimal to ASCII.

To LS:
- Convert ASCII to decimal.
- Map decimal to int32.

8.88.1 Example Corner Latitude Point 4 (Full) Conversion

For legacy purposes, both range-restricted (Tags 26-33) and full-range (Tag 82-89) representations of Image Corner Coordinates MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tags 82-89) being favored as per Section 6.3.

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth, should either not be reported, or be reported as an “error”.

Corner point 4 is the lower left corner of the captured image as highlighted in red (Figure 8-27).

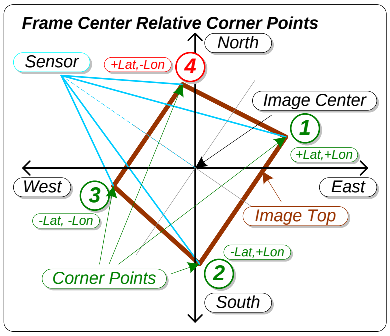

Figure 8-27: Offset Corner Point 4

Value is encoded using two’s complement.

8.89 Tag 89: Corner Longitude Point 4 (Full) Conversion

LS Tag	89	Units	Range	Format
LS Name	Corner Longitude Point 4 (Full)	Degrees	+/- 180	int32
US Mapped Key	Use EG0104 US Key			

Notes

- Frame Longitude for lower left corner.
- Full Range.
- Based on WGS84 ellipsoid.
- Map -(2^31-1)..(2^31-1) to +/-180.
- Use -(2^31) as an "error" indicator.
- -(2^31) = 0x80000000.
- Resolution: ~84 nano degrees.

Conversion Formula

$$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$$

$$LS\_89\_dec = \left(\frac{360}{4294967294} * LS\_89\right)$$

Example Value	Example LS Packet
29.1677346311172 Corrected Degrees	[K][L][V] = [0d89][0d2][0x10 CD]

US Key	06 0E 2B 34 01 01 01 03 07 01 02 01 03 0E 01 00	ESD Digraph	Rf
US Name	Corner Longitude Point 4 (Decimal Degrees)	ESD Name	SAR Longitude 3

Units	Range	Format	Units	Range	Format
Degrees	+/- 180	Double	Degrees	+/- 180.00	PDDDMMSST

Notes (US)

- Longitude coordinate of corner 4 of an image or bounding rectangle.
- Positive (+) is eastern hemisphere.
- Negative (-) is western hemisphere.

Notes (ESD)

- The longitude of the lower left corner of the SAR image box.

US Conversion

$$US\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$$

To US:
- US = (double)(360/0xFFFFFFFE * LS)

To LS:
- LS = (int32)round(0xFFFFFFFE/360 * US)

ESD Conversion

$$ESD\_dec = \left(\frac{360}{4294967294} * LS\_int\right)$$

To ESD:
- Convert LS to decimal.
- Convert decimal to ASCII.

To LS:
- Convert ASCII to decimal.
- Map decimal to int32.

8.89.1 Example Corner Longitude Point 4 (Full) Conversion

For legacy purposes, both range-restricted (Tags 26-33) and full-range (Tag 82-89) representations of Image Corner Coordinates MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tags 82-89) being favored as per Section 6.3.

The corner points of the captured image or image sequence have a real earth coordinate represented by a latitude-longitude pair. Corner points that lie above the horizon typically do not correspond to a point on the earth (an example being the tracking of an airborne object), should either not be reported, or be reported as an “error”.

Corner point 4 is the lower left corner of the captured image. See Figure 8-27 for Tag 88 above.

8.90 Tag 90: Platform Pitch Angle (Full) Conversion

LS Tag	90	**Units**	**Range**	**Format**
LS Name	Platform Pitch Angle (Full)	Degrees	+/- 90	int32
US Mapped Key	Use EG0104 US Key			

Notes	Conversion Formula
- Aircraft pitch angle. Angle between longitudinal axis and horizontal plane. - Positive angles above horizontal plane. - Map -(2^31-1)..(2^31-1) to +/-90. - Use -(2^31) as an "out of range" indicator. - -(2^31) = 0x80000000. - Res: ~42 nano deg.	US = (double)(180/0xFFFFFFFE * LS) $LS\_90\_dec = \left(\frac{180}{4294967294} * LS\_90\right)$

Example Value	Example LS Packet
-0.4315251 Degrees	[K][L][V] = [0d90][0d2][0xFD 3D]

US Key	06 0E 2B 34 01 01 01 07 07 01 10 01 05 00 00 00	ESD Digraph	Ip
US Name	Platform Pitch Angle	ESD Name	UAV Pitch (INS)

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Float	Degrees	+/- 20.00	PDD.HH

Notes	Notes
- Pitch angle of platform expressed in degrees. - The Pitch of an airborne platform describes the angle the longitudinal axis makes with the horizontal (i.e., equi-potential gravitational surface);	- Pitch angle of the aircraft.

US Conversion	ESD Conversion
$US\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$	$ESD\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$
To US: - US = (double)(180/0xFFFFFFFE * LS) *To LS:* - LS = (int32)round(0xFFFFFFFE/180 * US)	*To ESD:* - Convert LS to decimal. - Convert decimal to ASCII. *To LS:* - Convert ASCII to decimal. - Map decimal to int32.

8.90.1 Example Platform Pitch Angle (Full) Conversion

For legacy purposes, both range-restricted (Tag 6) and full-range (Tag 90) representations of Platform Pitch Angle MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tag 90) being favored as per Section 6.3.

The pitch angle of the platform is the angle between the longitudinal axis (line made by the fuselage) and the horizontal plane. Angles are positive when the platform nose is above the horizontal plane. This item allows unrestricted pitch angle values (see Figure 8-28).

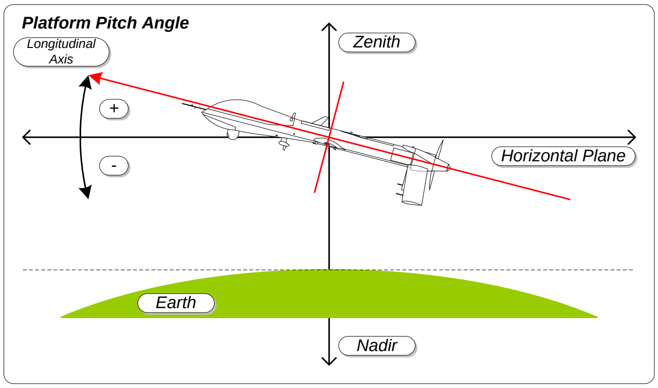

Figure 8-28: Platform Pitch Angle

Note that the int32 used in the LS value is encoded using two's complement.

8.91 Tag 91: Platform Roll Angle (Full) Conversion

LS Tag / LS Name / US Mapped Key		Units	Range	Format
LS Tag	91	Degrees	+/- 90	int32
LS Name	Platform Roll Angle (Full)			
US Mapped Key	Use EG0104 US Key			

Notes

- Platform roll angle. Angle between transverse axis and transvers-longitudinal plane. Positive angles for lowered right wing.
- Map -(2^31-1)..(2^31-1) to +/-90.
- Use -(2^31) as an "error" indicator.
- -(2^31) = 0x80000000.
- Resolution: ~42 nano degrees.

Conversion Formula

$$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$$

$$LS\_91\_dec = \left(\frac{180}{4294967294} * LS\_91\right)$$

Example Value	Example LS Packet
3.405814 Degrees	[K][L][V] = [0d91][0d2][0x08 B8]

US Key	06 0E 2B 34 01 01 01 07 07 01 10 01 04 00 00 00	ESD Digraph	Ir
US Name	Platform Roll Angle	ESD Name	UAV Roll (INS)

Units	Range	Format	Units	Range	Format
Degrees	+/- 90	Float	Degrees	+/- 50.00	PDD.HH

Notes (US)

- Roll angle of platform expressed in degrees.
- The Roll of an airborne platform is rotation about its longitudinal (front-to-back) axis;
- Wings level is zero degrees, positive (negative) angles describe a platform orientation with the right wing down(up).

Notes (ESD)

- Roll angle of the aircraft.

US Conversion

$$US\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$$

To US:
- US = (double)(180/0xFFFFFFFE * LS)

To LS:
- LS = (int32)round(0xFFFFFFFE/180 * US)

ESD Conversion

$$ESD\_dec = \left(\frac{180}{4294967294} * LS\_int\right)$$

To ESD:
- Convert LS to decimal.
- Convert decimal to ASCII.

To LS:
- Convert ASCII to decimal.
- Map decimal to int32.

8.91.1 Example Platform Roll Angle (Full) Conversion

For legacy purposes, both range-restricted (Tag 7) and full-range (Tag 91) representations of Platform Roll Angle MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tag 91) being favored as per Section 6.3.

The rotation operation performed about the longitudinal axis forms the roll angle between the previous aircraft transverse-longitudinal plane and the new transverse axis location (line from wing tip to wing tip). Positive angles correspond to the starboard (right) wing lowered below the previous aircraft transverse-longitudinal plane. This item allows unrestricted roll angles (see Figure 8-29).

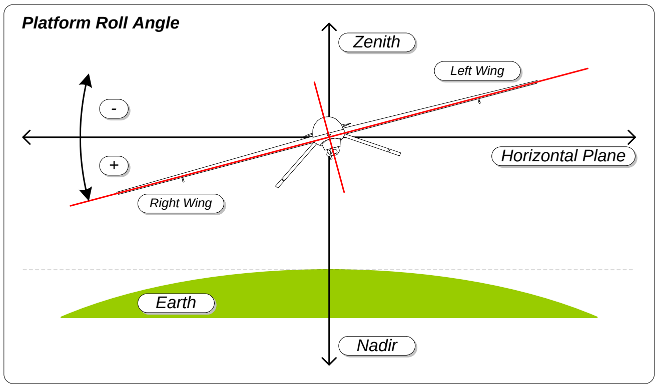

Figure 8-29: Platform Roll Angle

8.92 Tag 92: Platform Angle of Attack (Full) Conversion

LS Tag LS Name US Mapped Key	92 Platform Angle of Attack (Full) 06 0E 2B 34 01 01 01 01 0E 01 01 01 02 00 00 00	Units: Degrees	Range: +/- 90	Format: int32
Notes		**Conversion Formula**		
- Platform Attack Angle. Angle between platform longitudinal axis and relative wind. - Positive angles for upward relative wind. - Map -(2^31-1)..(2^31-1) to +/-90. - Use -(2^31) as an "out of range" indicator. - -(2^31) = 0x80000000. - Res: ~42 nano deg.		$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_92\_dec = \left(\frac{180}{4294967294} * LS\_92\right)$		
Example Value	**Example LS Packet**			
-8.670177 Degrees	[K][L][V] = [0d92][0d2][0xC8 83]			

8.92.1 Example Platform Angle of Attack (Full) Conversion

For legacy purposes, both range-restricted (Tag 50) and full-range (Tag 92) representations of Platform Angle of Attack MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tag 92) being favored as per Section 6.3.

The angle of attack of an airborne platform is the angle formed between the relative wind and platform longitudinal axis (line made by the fuselage). Positive angles for wind with a relative upward component. Refer to Figure 8-30.

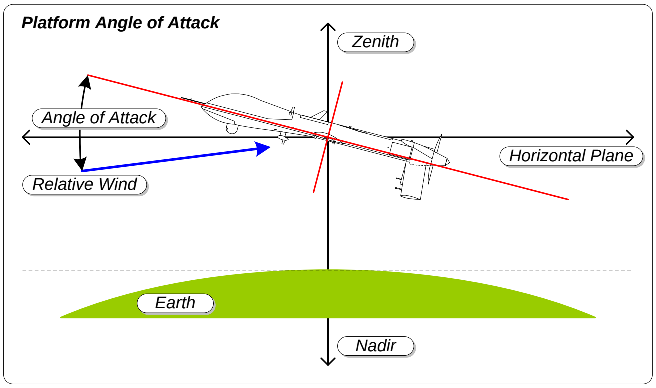

Figure 8-30: Platform Angle of Attack

Note that the int32 used in the LS value is encoded using two's complement.

8.93 Tag 93: Platform Sideslip Angle (Full) Conversion

LS Tag / LS Name / US Mapped Key	93 / Platform Sideslip Angle (Full) / 06 0E 2B 34 01 01 01 01 0E 01 01 01 04 00 00 00	Units	Range	Format
		Degrees	+/- 180	int32

Notes	Conversion Formula
- Angle between the platform longitudinal axis and relative wind. - Full Range. - Positive angles to right wing, neg to left. - Map -(2^31-1)..(2^31-1) to +/-90. - Use -(2^31) as an "out of range" indicator. - -(2^31) = 0x80000000. - Res: ~42 nano deg.	$LS\_dec = \left(\frac{LS\_range}{int\_range} * LS\_int\right)$ $LS\_93\_dec = \left(\frac{360}{4294967294} * LS\_93\right)$

Example Value	Example LS Packet
x	[K][L][V] = [0d93][0dx][x]

8.93.1 Example Platform Sideslip Angle (Full) Conversion

For legacy purposes, both range-restricted (Tag 52) and full-range (Tag 93) representations of Platform Sideslip Angle MAY appear in the same ST 0601 packet. A single representation is preferred, with the full-range version (Tag 93) being favored as per Section 6.3.

The angle formed between the platform longitudinal axis (line made by the fuselage) and the relative wind is the sideslip angle. A negative sideslip angle is depicted in Figure 8-31:

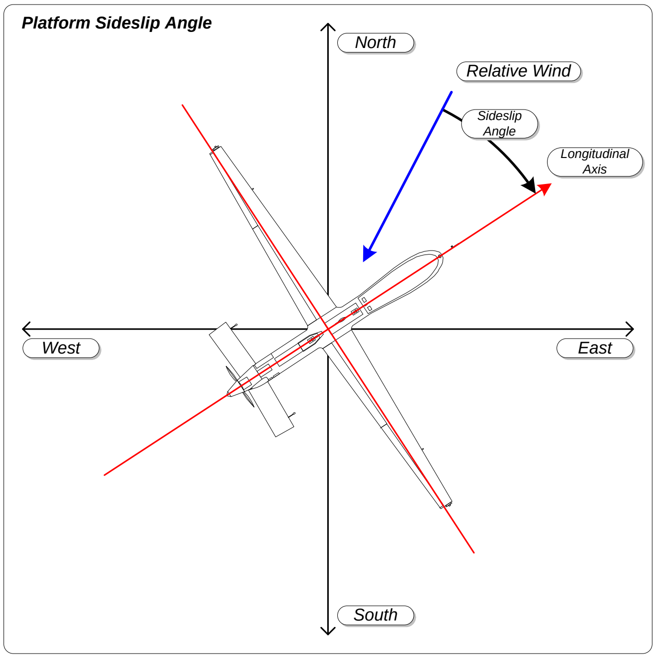

Figure 8-31: Platform Sideslip Angle

Note that the int32 used in the LS value is encoded using two's complement.

8.94 Tag 94: MIIS Core Identifier

		Units	Range	Format
LS Tag	94	None	None	Binary Value
LS Name	MIIS Core Identifier			
US Mapped Key	Use ST1204 MIIS Core 16-byte Key.			

Notes	Conversion Formula
- Local set tag to include the ST1204 MIIS Core Identifier Binary Value within ST0601. Use according to the rules and requirements defined in ST1204.	x x

Example Value	Example LS Packet
x	[K][L][V] = [0d94][0dx][x]

US Key	06 0E 2B 34 01 01 01 01 0E 01 04 05 03 00 00 00	ESD Digraph	x
US Name	Motion Imagery Identification System Core	ESD Name	x

Units	Range	Format	Units	Range	Format
None	None	Set	x	x	x

8.94.1 Example MIIS Core Identifier Details

ST 0601 Tag 94 allows users to include the MIIS Core Identifier (ST1204) Binary Value (opposed to the text-based representation) within ST 0601. Tag 94's value does not include ST1204's 16 byte Key or length, only the value portion.

See MISB ST 1204 [18] for generation and usage requirements.

8.95 Tag 95: SAR Motion Imagery Metadata

		Units	Range	Format
LS Tag	95	None	None	Set
LS Name	SAR Motion Imagery Metadata			
US Mapped Key	Use ST1206 SARMI 16-byte Key.			

Notes	Conversion Formula
- Local set tag to include the ST1206 SAR Motion Imagery Metadata Local Set data within ST0601. Use according to the rules and requirements defined in ST1206.	x x

Example Value	Example LS Packet
x	[K][L][V] = [0d95][0dx][x]

US Key	06 0E 2B 34 02 0B 01 01 0E 01 03 03 0D 00 00 00	ESD Digraph	x
US Name	SAR Motion Imagery Metadata	ESD Name	x

Units	Range	Format	Units	Range	Format
None	None	Set	x	x	x

8.95.1 Example SAR Motion Imagery Metadata Details

ST 0601 Tag 95 allows users to include the SAR Motion Imagery Metadata (ST1206) within ST 0601. The SARMI metadata set allows users to exploit both sequential synthetic aperture radar (SAR) imagery and sequential SAR cohenrent change products as Motion Imagery.

See MISB ST 1206 [19] for generation and usage requirements.

9 Appendix A – Deprecated Requirements

The following requirement was deprecated in ST 0601.6.

REQ-2.08 (ST 0601 decoders shall accept Universal Keys with any version number represented within byte 8.) as this is difficult to enforce from a compliance perspective, and is in with another requirement specifying the exact 16-byte KLV key to use (REQ-1.02) [REQ-1.02 is now REQ. ST 0601.8-18].