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.

[5] MISB ST 0604.3 Time Stamping Compressed Motion Imagery, Feb 2014
ST 0601.8 UAS Datalink Local Set

[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
[10] MISB ST 0806.4 Remote Video Terminal Local set, Feb 2014
[12] MISB ST 0902.3 Motion Imagery Sensor Minimum Metadata Set, Feb 2014
[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

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Summary of Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST 0601.8</td>
<td>6/18/2014</td>
<td>• EARS requirements format and other general formatting</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Changed LDS to LS and UDS to US</td>
</tr>
</tbody>
</table>

23 October 2014 Motion Imagery Standards Board 2
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.

<table>
<thead>
<tr>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST 0601.8-01</td>
</tr>
<tr>
<td>ST 0601.8-02</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST 0601.8-03</td>
</tr>
<tr>
<td>ST 0601.8-04</td>
</tr>
<tr>
<td>ST 0601.8-05</td>
</tr>
<tr>
<td>ST 0601.8-06</td>
</tr>
<tr>
<td>ST 0601.8-07</td>
</tr>
<tr>
<td>ST 0601.8-08</td>
</tr>
</tbody>
</table>

6.2 Mandatory UAS Datalink LS items

<table>
<thead>
<tr>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST 0601.8-09</td>
</tr>
<tr>
<td>ST 0601.8-10</td>
</tr>
<tr>
<td>ST 0601.8-11</td>
</tr>
<tr>
<td>ST 0601.8-12</td>
</tr>
</tbody>
</table>

6.3 Metadata Usage

<table>
<thead>
<tr>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST 0601.8-13</td>
</tr>
<tr>
<td>ST 0601.8-14</td>
</tr>
<tr>
<td>ST 0601.8-15</td>
</tr>
<tr>
<td>ST 0601.8-16</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ST 0601.8-18</strong></td>
</tr>
</tbody>
</table>

**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. |

<table>
<thead>
<tr>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ST 0601.8-19</strong></td>
</tr>
</tbody>
</table>

### 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.

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:

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:

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 byte, 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.
### 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 6-7:** Nested Packet Example

**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 idea 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-platform 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 requirements 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 requirements 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:

![Packet Timestamp Example](image)

**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:

![Checksum Computation Range](image)

**Figure 6-10: Checksum Computation Range**

An example algorithm for calculating the checksum is shown below:

```c
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

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Checksum</td>
<td>06 0E 28 34 01 01 01 01 0E 01 02 03 01 00 00 00 (CRC 56132)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>None</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
</tr>
<tr>
<td>2</td>
<td>UNIX Time Stamp</td>
<td>Use EG0104 US Key</td>
<td>x</td>
<td>x</td>
<td>06 0E 28 34 01 01 01 03 07 02 01 01 01 05 00 00 (CRC 64827)</td>
<td>User Defined Time Stamp – microseconds since 1970</td>
<td>Microseconds</td>
<td>uint64</td>
<td>8</td>
<td>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.</td>
</tr>
<tr>
<td>3</td>
<td>Mission ID</td>
<td>06 0E 28 34 01 01 01 01 0E 01 04 01 03 00 00 00 (CRC 65358)</td>
<td>Mn</td>
<td>Mission Number</td>
<td>06 0E 28 34 01 01 01 01 01 05 05 00 00 00 00 (CRC 37735)</td>
<td>Episode Number</td>
<td>String</td>
<td>ISO 646</td>
<td>V</td>
<td>Descriptive Mission Identifier to distinguish event or sortie. Value field is Free Text. Maximum 127 characters.</td>
</tr>
<tr>
<td>4</td>
<td>Platform Tail Number</td>
<td>06 0E 28 34 01 01 01 01 0E 01 04 01 02 00 00 00 (CRC 35322)</td>
<td>Pt</td>
<td>Platform Tail Number</td>
<td>x</td>
<td>x</td>
<td>String</td>
<td>ISO 646</td>
<td>V</td>
<td>Identifier of platform as posted. E.g.: &quot;AF008&quot;, &quot;BP101&quot;, etc. Value field is Free Text. Maximum 127 characters.</td>
</tr>
<tr>
<td>5</td>
<td>Platform Heading Angle</td>
<td>Use EG0104 US Key</td>
<td>Ih</td>
<td>UAV Heading (INS)</td>
<td>06 0E 28 34 01 01 01 07 07 01 10 01 06 00 00 00 (CRC 23727)</td>
<td>Platform Heading Angle</td>
<td>Degrees</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
</tr>
<tr>
<td>6</td>
<td>Platform Pitch Angle</td>
<td>Use EG0104 US Key</td>
<td>Ip</td>
<td>UAV Pitch (INS)</td>
<td>06 0E 28 34 01 01 01 07 07 01 10 01 05 00 00 00 (CRC 51059)</td>
<td>Platform Pitch Angle</td>
<td>Degrees</td>
<td>int16</td>
<td>2</td>
<td>Aircraft pitch angle. Angle between longitudinal axis and horizontal plane. Positive angles above horizontal plane. Map –(2^15) to +/-20. Use –(2^15) as &quot;out of range&quot; indicator. –(2^15) = 0x8000. Resolution: ~610 micro degrees.</td>
</tr>
<tr>
<td>7</td>
<td>Platform Roll Angle</td>
<td>Use EG0104 US Key</td>
<td>Ir</td>
<td>UAV Roll (INS)</td>
<td>06 0E 28 34 01 01 01 07 07 01 10 01 04 00 00 00 (CRC 45511)</td>
<td>Platform Roll Angle</td>
<td>Degrees</td>
<td>int16</td>
<td>2</td>
<td>Platform roll angle. Angle between transverse axis and transvers–longitudinal plane. Positive angles for lowered right wing. Map –(2^15) to +/-50. Use –(2^15) as &quot;out of range&quot; indicator. –(2^15) = 0x8000. Res.: ~1525 micro deg.</td>
</tr>
<tr>
<td>8</td>
<td>Platform True Airspeed</td>
<td>06 0E 28 34 01 01 01 01 0E 01 01 01 0A 00 00 00 (CRC 20280)</td>
<td>As</td>
<td>True Airspeed</td>
<td>x</td>
<td>x</td>
<td>Meters/Second</td>
<td>uint8</td>
<td>1</td>
<td>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.</td>
</tr>
<tr>
<td>9</td>
<td>Platform Indicated Airspeed</td>
<td>06 0E 28 34 01 01 01 01 0E 01 01 01 0B 00 00 00 (CRC 14732)</td>
<td>Ai</td>
<td>Indicated Airspeed</td>
<td>x</td>
<td>x</td>
<td>Meters/Second</td>
<td>uint8</td>
<td>1</td>
<td>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.</td>
</tr>
<tr>
<td>TAG</td>
<td>LS Name</td>
<td>Mapped US</td>
<td>ESD</td>
<td>ESD Name</td>
<td>US</td>
<td>US Name</td>
<td>Units</td>
<td>Format</td>
<td>Len</td>
<td>Notes</td>
</tr>
<tr>
<td>-----</td>
<td>---------</td>
<td>-----------</td>
<td>-----</td>
<td>----------</td>
<td>----</td>
<td>---------</td>
<td>-------</td>
<td>--------</td>
<td>-----</td>
<td>-------</td>
</tr>
<tr>
<td>10</td>
<td>Platform Designation</td>
<td>Use EG0104 US Key</td>
<td>Pc</td>
<td>Project ID Code</td>
<td>06 0E 2B 34 01 01 01 01 01 01 01 20 01 00 00 00 00 (CRC 36601)</td>
<td>Device Designation</td>
<td>String</td>
<td>ISO 646</td>
<td>V</td>
<td>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.</td>
</tr>
<tr>
<td>11</td>
<td>Image Source Sensor</td>
<td>Use EG0104 US Key</td>
<td>Sn</td>
<td>Sensor Name</td>
<td>06 0E 2B 34 01 01 01 01 01 01 01 20 01 01 01 00 00 00 00 (CRC 53038)</td>
<td>Image Source Device</td>
<td>String</td>
<td>ISO 646</td>
<td>V</td>
<td>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.</td>
</tr>
<tr>
<td>12</td>
<td>Image Coordinate System</td>
<td>Use EG0104 US Key</td>
<td>Ic</td>
<td>Image Coordinate System</td>
<td>06 0E 2B 34 01 01 01 01 01 01 01 20 01 01 01 00 00 00 00 (CRC 32410)</td>
<td>Image Coordinate System</td>
<td>String</td>
<td>ISO 646</td>
<td>V</td>
<td>String of the image coordinate system used. E.g.: 'Geodetic WGS84', 'Geocentric WGS84', 'UTM', 'None', etc. Value field is Free Text. Maximum 127 characters.</td>
</tr>
<tr>
<td>13</td>
<td>Sensor Latitude</td>
<td>Use EG0104 US Key</td>
<td>Sa</td>
<td>Sensor Latitude</td>
<td>06 0E 2B 34 01 01 01 01 01 01 01 20 01 02 01 02 00 00 00 00 (CRC 8663)</td>
<td>Device Latitude</td>
<td>Degrees</td>
<td>int32</td>
<td>4</td>
<td>Sensor Latitude. Based on WGS84 ellipsoid. Map -2^(31) to +90. Use -2^(31) as an &quot;error&quot; indicator. -2^(31) = 0x80000000. Resolution: ~42 nano degrees.</td>
</tr>
<tr>
<td>14</td>
<td>Sensor Longitude</td>
<td>Use EG0104 US Key</td>
<td>So</td>
<td>Sensor Longitude</td>
<td>06 0E 2B 34 01 01 01 01 01 01 01 20 01 02 01 02 00 00 00 00 (CRC 20407)</td>
<td>Device Longitude</td>
<td>Degrees</td>
<td>int32</td>
<td>4</td>
<td>Sensor Longitude. Based on WGS84 ellipsoid. Map -2^(31) to +180. Use -2^(31) as an &quot;error&quot; indicator. -2^(31) = 0x80000000. Resolution: ~84 nano degrees.</td>
</tr>
<tr>
<td>15</td>
<td>Sensor True Altitude</td>
<td>Use EG0104 US Key</td>
<td>Sl</td>
<td>Sensor Altitude</td>
<td>06 0E 2B 34 01 01 01 01 01 01 01 20 01 01 01 07 01 02 01 02 02 00 00 00 (CRC 13170)</td>
<td>Device Altitude</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>Altitude of sensor as measured from Mean Sea Level (MSL). Map 0..2^16 to -900..19000 meters. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.</td>
</tr>
<tr>
<td>16</td>
<td>Sensor Horizontal field of View</td>
<td>Use EG0104 US Key</td>
<td>Fv</td>
<td>field of View</td>
<td>06 0E 2B 34 01 01 01 01 01 01 01 20 01 02 01 02 01 08 00 00 00 (CRC 23753)</td>
<td>field of View (FOV–Horizontal)</td>
<td>Degrees</td>
<td>uint16</td>
<td>2</td>
<td>Horizontal field of view of selected imaging sensor. Map 0..2^16 to 0..180. Resolution: ~2.7 milli degrees.</td>
</tr>
<tr>
<td>17</td>
<td>Sensor Vertical Field of View</td>
<td>06 0E 2B 34 01 01 01 07 04 20 02 01 01 0A 01 00 (CRC 30292)</td>
<td>Vv</td>
<td>Vertical Field of View</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>uint16</td>
<td>2</td>
<td>Vertical field of view of selected imaging sensor. Map 0..2^16 to 0..180. Resolution: ~2.7 milli degrees. Requires data conversion between LS value and SMPTE Mapped US Key.</td>
</tr>
<tr>
<td>18</td>
<td>Sensor Relative Azimuth Angle</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 0E 01 01 02 04 00 00 00 (CRC 944)</td>
<td>Az</td>
<td>Sensor Relative Azimuth Angle</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>uint32</td>
<td>4</td>
<td>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 to 0..360. Resolution: ~84 nano degrees.</td>
</tr>
<tr>
<td>19</td>
<td>Sensor Relative Elevation Angle</td>
<td>06 0E 2B 34 01 01 01 01</td>
<td>De</td>
<td>Sensor Relative</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>int32</td>
<td>4</td>
<td>Relative Elevation Angle of sensor to platform longitudinal–transverse plane. Negative angles down.</td>
</tr>
<tr>
<td>TAG</td>
<td>LS Name</td>
<td>Mapped US</td>
<td>ESD</td>
<td>ESD Name</td>
<td>US</td>
<td>US Name</td>
<td>Units</td>
<td>Format</td>
<td>Len</td>
<td>Notes</td>
</tr>
<tr>
<td>------</td>
<td>--------------------------</td>
<td>-----------</td>
<td>-----</td>
<td>----------------</td>
<td>------</td>
<td>---------------</td>
<td>-------</td>
<td>--------</td>
<td>----</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>20</td>
<td>Sensor Relative Roll Angle</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 02 06 00 00 00 (CRC 61144)</td>
<td>Ro</td>
<td>Sensor Relative Roll Angle</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>uint32</td>
<td>4</td>
<td>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.</td>
</tr>
<tr>
<td>21</td>
<td>Slant Range</td>
<td>Use EG0104 US Key</td>
<td>Sr</td>
<td>Slant Range</td>
<td>06 0E 2B 34 01 01 01 01 07 01 08 01 01 00 00 00 (CRC 16588)</td>
<td>Slant Range</td>
<td>Meters</td>
<td>uint32</td>
<td>4</td>
<td>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.</td>
</tr>
<tr>
<td>22</td>
<td>Target Width</td>
<td>Use EG0104 US Key</td>
<td>Tw</td>
<td>Target Width</td>
<td>06 0E 2B 34 01 01 01 01 07 01 09 02 01 00 00 00 (CRC 60350)</td>
<td>Target Width</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>Target Width within sensor field of view. Map 0..(2^16–1) to 0..10000 meters. 1 meter = 3.2808399 feet. Resolution: ~.16 meters.</td>
</tr>
<tr>
<td>23</td>
<td>Frame Center Latitude</td>
<td>Use EG0104 US Key</td>
<td>Ta</td>
<td>Frame Center Latitude</td>
<td>Ta</td>
<td>Frame Center Latitude</td>
<td>Degrees</td>
<td>uint32</td>
<td>4</td>
<td>Terrain Latitude of frame center. Based on WGS84 ellipsoid. Map (2^31–1) to 0..+/-90. Use -2(31) as an &quot;error&quot; indicator. -(2^31) = 0x80000000. Resolution: ~42 nano degrees.</td>
</tr>
<tr>
<td>24</td>
<td>Frame Center Longitude</td>
<td>Use EG0104 US Key</td>
<td>To</td>
<td>Frame Center Longitude</td>
<td>To</td>
<td>Frame Center Longitude</td>
<td>Degrees</td>
<td>uint32</td>
<td>4</td>
<td>Terrain Longitude of frame center. Based on WGS84 ellipsoid. Map (2^31–1) to 0..+/-180. Use -(2^31) as an &quot;error&quot; indicator. -(2^31) = 0x80000000. Resolution: ~84 nano degrees.</td>
</tr>
<tr>
<td>25</td>
<td>Frame Center Elevation</td>
<td>06 0E 2B 34 01 01 01 01 07 01 02 01 03 16 00 00 (CRC 57054)</td>
<td>Te</td>
<td>Frame Center Elevation</td>
<td>x</td>
<td>x</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>Terrain elevation at frame center relative to Mean Sea Level (MSL). Map 0..(2^16–1) to 900..19000 meters. Resolution: ~0.3 meters.</td>
</tr>
<tr>
<td>26</td>
<td>Offset Corner Latitude Point 1</td>
<td>Use EG0104 US Key</td>
<td>Rg</td>
<td>SAR Latitude 4</td>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 07 01 00 (CRC 23392)</td>
<td>Corner Latitude Point 1 (Decimal Degrees)</td>
<td>Degrees</td>
<td>uint16</td>
<td>2</td>
<td>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..+/-0.075. Use -(2^15) as an &quot;error&quot; indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.</td>
</tr>
<tr>
<td>27</td>
<td>Offset Corner Longitude Point 1</td>
<td>Use EG0104 US Key</td>
<td>Rh</td>
<td>SAR Longitude 4</td>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 08 01 00 (CRC 11777)</td>
<td>Corner Longitude Point 1 (Decimal Degrees)</td>
<td>Degrees</td>
<td>uint16</td>
<td>2</td>
<td>Frame Longitude, offset for upper left corner. Based on WGS84 ellipsoid. Use with Frame Center Latitude. Map -(2^15–1..2^15–1) to 0..+/-0.075. Use -(2^15) as an &quot;error&quot; indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.</td>
</tr>
<tr>
<td>28</td>
<td>Offset Corner Latitude Point 2</td>
<td>Use EG0104 US Key</td>
<td>Ra</td>
<td>SAR Latitude 1</td>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 08 01 00</td>
<td>Corner Latitude Point 2 (Decimal Degrees)</td>
<td>Degrees</td>
<td>uint16</td>
<td>2</td>
<td>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..+/-0.075.</td>
</tr>
<tr>
<td>TAG</td>
<td>LS Name</td>
<td>Mapped US</td>
<td>ESD</td>
<td>ESD Name</td>
<td>US</td>
<td>US Name</td>
<td>Units</td>
<td>Format</td>
<td>Len</td>
<td>Notes</td>
</tr>
<tr>
<td>-----</td>
<td>---------</td>
<td>-----------</td>
<td>-----</td>
<td>----------</td>
<td>----</td>
<td>---------</td>
<td>-------</td>
<td>--------</td>
<td>-----</td>
<td>-------</td>
</tr>
<tr>
<td>29</td>
<td>Offset Corner Longitude Point 2</td>
<td>Use EG0104 US Key</td>
<td>Rb</td>
<td>SAR Longitude 1</td>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 0C 01 00 (CRC 43921)</td>
<td>Corner Longitude Point 2 (Decimal Degrees)</td>
<td>Degrees</td>
<td>int16</td>
<td>2</td>
<td>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 &quot;error&quot; indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.</td>
</tr>
<tr>
<td>30</td>
<td>Offset Corner Latitude Point 3</td>
<td>Use EG0104 US Key</td>
<td>Rc</td>
<td>SAR Latitude 2</td>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 09 01 00 (CRC 16481)</td>
<td>Corner Latitude Point 3 (Decimal Degrees)</td>
<td>Degrees</td>
<td>int16</td>
<td>2</td>
<td>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 &quot;error&quot; indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.</td>
</tr>
<tr>
<td>31</td>
<td>Offset Corner Longitude Point 3</td>
<td>Use EG0104 US Key</td>
<td>Rd</td>
<td>SAR Longitude 2</td>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 00 00 00 (CRC 40097)</td>
<td>Corner Longitude Point 3 (Decimal Degrees)</td>
<td>Degrees</td>
<td>int16</td>
<td>2</td>
<td>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 &quot;error&quot; indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.</td>
</tr>
<tr>
<td>32</td>
<td>Offset Corner Latitude Point 4</td>
<td>Use EG0104 US Key</td>
<td>Re</td>
<td>SAR Latitude 3</td>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 0A 01 00 (CRC 6449)</td>
<td>Corner Latitude Point 4 (Decimal Degrees)</td>
<td>Degrees</td>
<td>int16</td>
<td>2</td>
<td>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 &quot;error&quot; indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.</td>
</tr>
<tr>
<td>33</td>
<td>Offset Corner Longitude Point 4</td>
<td>Use EG0104 US Key</td>
<td>Rf</td>
<td>SAR Longitude 3</td>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 0E 01 00 (CRC 50673)</td>
<td>Corner Longitude Point 4 (Decimal Degrees)</td>
<td>Degrees</td>
<td>int16</td>
<td>2</td>
<td>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 &quot;error&quot; indicator. -(2^15) = 0x8000. Resolution: ~1.2micro deg, ~0.25meters at equator.</td>
</tr>
<tr>
<td>34</td>
<td>Icing Detected</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 0C 00 00 00 (CRC 26785)</td>
<td>Id</td>
<td>Icing Detected</td>
<td>x</td>
<td>x</td>
<td>Icing Code</td>
<td>uint8</td>
<td>1</td>
<td>Flag for icing detected at aircraft location. 0: Detector off 1: No icing Detected 2: Icing Detected</td>
</tr>
<tr>
<td>35</td>
<td>Wind Direction</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 0D 00 00 00 (CRC 7201)</td>
<td>Wd</td>
<td>Wind Direction</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
</tr>
<tr>
<td>36</td>
<td>Wind Speed</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 0E 00 00 00</td>
<td>Ws</td>
<td>Wind Speed</td>
<td>x</td>
<td>x</td>
<td>Meters/Second</td>
<td>uint8</td>
<td>1</td>
<td>Wind speed at aircraft location. Map 0..255 to 0..100 meters/second. 1 m/s = 1.94384449 knots. Resolution: ~0.4 meters / second.</td>
</tr>
</tbody>
</table>
### Static Pressure

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>37</td>
<td>Static Pressure</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 0F 00 00 00 (CRC 62333)</td>
<td>Ps</td>
<td>Static Pressure</td>
<td>x</td>
<td>Millbar</td>
<td>uint16</td>
<td>2</td>
<td>Static pressure at aircraft location. Map 0..(2^16-1) to 0..5000 mbar. 1 mbar = 0.0145037738 PSI. Resolution: ~0.08 Millibar</td>
<td></td>
</tr>
</tbody>
</table>

### Density Altitude

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>Density Altitude</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 10 00 00 00 (CRC 15412)</td>
<td>Da</td>
<td>Density Altitude</td>
<td>x</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
<td></td>
</tr>
</tbody>
</table>

### Outside Air Temperature

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>39</td>
<td>Outside Air Temperature</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 11 00 00 00 (CRC 19072)</td>
<td>At</td>
<td>Air Temperature</td>
<td>x</td>
<td>Celcius</td>
<td>int8</td>
<td>1</td>
<td>Temperature outside of aircraft. -128..127 Degrees Celsius. Resolution: 1 degree celsius.</td>
<td></td>
</tr>
</tbody>
</table>

### Target Location Latitude

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>Target Location Latitude</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 02 00 00 00 (CRC 36472)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>int32</td>
<td>4</td>
<td>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.</td>
</tr>
</tbody>
</table>

### Target Location Longitude

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>41</td>
<td>Target Location Longitude</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 03 03 00 00 00 (CRC 63692)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>int32</td>
<td>4</td>
<td>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.</td>
</tr>
</tbody>
</table>

### Target Location Elevation

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>Target Location Elevation</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 03 04 00 00 00 (CRC 43489)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
</tr>
</tbody>
</table>

### Target Track Gate Width

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>43</td>
<td>Target Track Gate Width</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 03 05 00 00 00 (CRC 52173)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Pixels</td>
<td>uint8</td>
<td>1</td>
<td>Tracking gate width (x value) of tracked target within field of view. Closely tied to source video resolution in pixels.</td>
</tr>
</tbody>
</table>

### Target Track Gate Height

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>44</td>
<td>Target Track Gate Height</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 03 06 00 00 00 (CRC 17545)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Pixels</td>
<td>uint8</td>
<td>1</td>
<td>Tracking gate height (y value) of tracked target within field of view. Closely tied to source video resolution in pixels.</td>
</tr>
</tbody>
</table>

### Target Error Estimate – CE90

<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>45</td>
<td>Target Error Estimate – CE90</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 03 07 00 00 00 (CRC 17545)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>Circular Error 90 (CE90) is the estimated error distance in the horizontal direction.</td>
</tr>
<tr>
<td>TAG</td>
<td>LS Name</td>
<td>Mapped US</td>
<td>ESD</td>
<td>ESD Name</td>
<td>US</td>
<td>US Name</td>
<td>Units</td>
<td>Format</td>
<td>Len</td>
<td>Notes</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------</td>
<td>-----------------</td>
<td>-----</td>
<td>----------</td>
<td>-------</td>
<td>-------------</td>
<td>-------</td>
<td>--------</td>
<td>----</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>46</td>
<td>Target Error Estimate – LE90</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 03 08 00 00 00 (CRC 59091)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>Specifies the radius of 90% probability on a plane tangent to the earth’s surface. Res: ~0.0624 meters</td>
</tr>
<tr>
<td>47</td>
<td>Generic Flag Data 01</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 03 01 00 00 00 (CRC 5540)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>None</td>
<td>uint8</td>
<td>1</td>
<td>Generic Flagged Metadata Position Format msb8..lsb 1- Laser Range 1on,0off 2- Auto-Track 1on,0off 3- IR Polarity 1blk,0wht 4- Iceing detected 1Ice,0(off/no ice) 5- Slant Range 1measured, 0calc 6- Image Invalid 1invalid, 0valid 7,8- Use 0</td>
</tr>
<tr>
<td>48</td>
<td>Security Local Metadata Set</td>
<td>Use ST0102 US key for Local Sets.</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Security Local Metadata Set</td>
<td>None</td>
<td>Set</td>
<td>x</td>
</tr>
<tr>
<td>49</td>
<td>Differential Pressure</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 01 00 00 00 (CRC 20775)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Millibar</td>
<td>uint16</td>
<td>2</td>
<td>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</td>
</tr>
<tr>
<td>50</td>
<td>Platform Angle of Attack</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 01 00 00 00 (CRC 51963)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>int16</td>
<td>2</td>
<td>Platform Attack Angle. Angle between platform longitudinal axis and relative wind. Positive angles for upward relative wind. Map −(2^15−1)..&lt;2^15−1) to +/-20. Use –(2^15) as an “out of range” indicator. −(2^15) = 0x8000. Res: ~610 micro degrees.</td>
</tr>
<tr>
<td>51</td>
<td>Platform Vertical Speed</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 03 00 00 00 (CRC 48207)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Meters/Second</td>
<td>int16</td>
<td>2</td>
<td>Vertical speed of the aircraft relative to zenith. Positive ascending, negative descending. Map−(2^15−1)..&lt;2^15−1) to +/-180. Use –(2^15) as an “out of range” indicator. −(2^15) = 0x8000. Resolution: ~ 0.0055 meters/second.</td>
</tr>
<tr>
<td>52</td>
<td>Platform Sideslip Angle</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 04 00 00 00 (CRC 60770)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>int16</td>
<td>2</td>
<td>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)..&lt;2^15−1) to +/-20. Use –(2^15) as an “out of range” indicator. −(2^15) = 0x8000.</td>
</tr>
<tr>
<td>TAG</td>
<td>LS Name</td>
<td>Mapped US</td>
<td>ESD</td>
<td>ESD Name</td>
<td>US</td>
<td>US Name</td>
<td>Units</td>
<td>Format</td>
<td>Len</td>
<td>Notes</td>
</tr>
<tr>
<td>---------</td>
<td>----------------------------------</td>
<td>-----------------</td>
<td>-----</td>
<td>-----------</td>
<td>----</td>
<td>---------</td>
<td>-------</td>
<td>--------</td>
<td>-----</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>53</td>
<td>Airfield Barometric Pressure</td>
<td>06 0E 2B 34</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>Millbar</td>
<td>uint16</td>
<td>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</td>
</tr>
<tr>
<td>54</td>
<td>Airfield Elevation</td>
<td>06 0E 2B 34</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>Meters</td>
<td>uint16</td>
<td>2 Elevation of Airfield corresponding to Airfield Barometric Pressure. Map 0..(2^16–1) to 0..19000 meters. Offset = 0.1m. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.</td>
</tr>
<tr>
<td>55</td>
<td>Relative Humidity</td>
<td>06 0E 2B 34</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>Percent</td>
<td>uint8</td>
<td>1 Relative Humidity at aircraft location. Map 0..(2^8–1) to 0..100. Resolution: ~0.4%.</td>
</tr>
<tr>
<td>56</td>
<td>Platform Ground Speed</td>
<td>06 0E 2B 34</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>Meters/Second</td>
<td>uint8</td>
<td>Speed projected to the ground of an airborne platform passing overhead. 0..255 meters/second. 1 m/s = 1.94384449 knots. Resolution: 1 meter/second.</td>
</tr>
<tr>
<td>57</td>
<td>Ground Range</td>
<td>06 0E 2B 34</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>Meters</td>
<td>uint32</td>
<td>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.</td>
</tr>
<tr>
<td>58</td>
<td>Platform Fuel Remaining</td>
<td>06 0E 2B 34</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>Kilogram</td>
<td>uint16</td>
<td>2 Remaining fuel on airborne platform. Metered as fuel weight remaining. Map 0..(2^16–1) to 0..1000000 Kilograms. 1 kilogram = 2.20462262 pounds. Resolution: ~16 kilograms.</td>
</tr>
<tr>
<td>59</td>
<td>Platform Call Sign</td>
<td>06 0E 2B 34</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>String</td>
<td>ISO 646</td>
<td>V Call Sign of platform or operating unit. Value field is Free Text.</td>
</tr>
<tr>
<td>60</td>
<td>Weapon Load</td>
<td>06 0E 2B 34</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>uint16</td>
<td>nibble</td>
<td>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</td>
</tr>
<tr>
<td>61</td>
<td>Weapon Fired</td>
<td>06 0E 2B 34</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>uint8</td>
<td>nibble</td>
<td>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</td>
</tr>
<tr>
<td>TAG</td>
<td>LS Name</td>
<td>Mapped US</td>
<td>ESD</td>
<td>ESD Name</td>
<td>US</td>
<td>US Name</td>
<td>Units</td>
<td>Format</td>
<td>Len</td>
<td>Notes</td>
</tr>
<tr>
<td>-----</td>
<td>---------</td>
<td>-----------</td>
<td>-----</td>
<td>----------</td>
<td>----</td>
<td>---------</td>
<td>-------</td>
<td>--------</td>
<td>-----</td>
<td>-------</td>
</tr>
<tr>
<td>62</td>
<td>Laser PRF Code</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 02 02 01 00 00 00 (CRC 28949)</td>
<td>Lc</td>
<td>Laser PRF Code</td>
<td>x</td>
<td>x</td>
<td>None</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
</tr>
<tr>
<td>63</td>
<td>Sensor Field of View Name</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 02 02 02 00 00 00 (CRC 60105)</td>
<td>Vn</td>
<td>Sensor Field of View Name</td>
<td>x</td>
<td>x</td>
<td>List</td>
<td>uint8</td>
<td>1</td>
<td>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</td>
</tr>
<tr>
<td>64</td>
<td>Platform Magnetic Heading</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 08 00 00 00 (CRC 41552)</td>
<td>Mh</td>
<td>Platform Magnetic Heading</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
</tr>
<tr>
<td>65</td>
<td>UAS LS Version Number</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 02 03 03 00 00 00 (CRC 13868)</td>
<td>Iv</td>
<td>ESD ICD Version</td>
<td>x</td>
<td>x</td>
<td>Number</td>
<td>uint8</td>
<td>1</td>
<td>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.</td>
</tr>
<tr>
<td>66</td>
<td>Target Location Covariance Matrix</td>
<td>06 0E 2B 34 02 05 01 01 0E 01 03 03 14 00 00 00 (CRC 28126)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>TBD</td>
<td>TBD</td>
<td>TBD</td>
<td>Covariance Matrix of the error associated with a targeted location. Details TBD.</td>
</tr>
<tr>
<td>67</td>
<td>Alternate Platform Latitude</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 14 00 00 00 (CRC 63173)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>int32</td>
<td>4</td>
<td>Alternate Platform Latitude. Represents latitude of platform connected with UAS. Based on WGS84 ellipsoid. Map -2(2^31-1),2(2^31-1) to +/-90. Use -2(2^31) as an “error” indicator. -2(2^31) = 0x80000000. Resolution: ~42 nano degrees.</td>
</tr>
<tr>
<td>68</td>
<td>Alternate Platform Longitude</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 15 00 00 00 (CRC 32881)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>int32</td>
<td>4</td>
<td>Alternate Platform Longitude. Represents longitude of platform connected with UAS. Based on WGS84 ellipsoid. Map -2(2^31-1),2(2^31-1) to +/-180. Use -2(2^31) as an “error” indicator. -2(2^31) = 0x80000000. Resolution: ~84 nano degrees.</td>
</tr>
<tr>
<td>69</td>
<td>Alternate Platform Altitude</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 16 00 00 00 (CRC 7085)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>Altitude of alternate platform as measured from Mean Sea Level (MSL). Represents altitude of platform connected with UAS. Map 0..(2^16-1) to -900..19000 meters. 1 meter = 3.2808399 feet. Resolution: ~0.3 meters.</td>
</tr>
<tr>
<td>TAG</td>
<td>LS Name</td>
<td>Mapped US</td>
<td>ESD</td>
<td>ESD Name</td>
<td>US</td>
<td>US Name</td>
<td>Units</td>
<td>Format</td>
<td>Len</td>
<td>Notes</td>
</tr>
<tr>
<td>-----</td>
<td>-------------------------------</td>
<td>------------------------------------</td>
<td>-----</td>
<td>----------</td>
<td>----</td>
<td>---------</td>
<td>-------</td>
<td>--------</td>
<td>-----</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>70</td>
<td>Alternate Platform Name</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 17 00 00 00 (CRC 27929)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>String</td>
<td>ISO 646</td>
<td>V</td>
<td>Name of alternate platform connected to UAS. E.g.: 'Apache', 'Rover', 'Predator', 'Reaper', 'Outrider', 'Pioneer', 'IgnatER', 'Warrior', 'Shadow', 'Hunter II', 'Global Hawk', 'Scan Eagle', etc. Value field is Free Text. Maximum 127 characters.</td>
</tr>
<tr>
<td>71</td>
<td>Alternate Platform Heading</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 18 00 00 00 (CRC 47607)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Degrees</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
</tr>
<tr>
<td>72</td>
<td>Event Start Time – UTC</td>
<td>Use EG0104 US Key</td>
<td>x</td>
<td>Mission Start Time, Date, and Date of Collection</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 02 01 02 07 01 00 00 (CRC 11991)</td>
<td>Event Start Date Time – UTC</td>
<td>Microseconds</td>
<td>uint64</td>
<td>8</td>
<td>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.</td>
</tr>
<tr>
<td>73</td>
<td>RVT Local Set</td>
<td>Use ST0806 RVT LS 16-byte Key.</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Remote Video Terminal Local Set</td>
<td>None</td>
<td>Set</td>
<td>x</td>
</tr>
<tr>
<td>74</td>
<td>VMTI Data Set</td>
<td>Use ST0903 VMTI LS 16-byte Key.</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Video Moving Target Indicator Local Set</td>
<td>None</td>
<td>Set</td>
<td>x</td>
</tr>
<tr>
<td>75</td>
<td>Sensor Ellipsoid Height</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 02 01 82 47 00 00 (CRC 16670)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
</tr>
<tr>
<td>76</td>
<td>Alternate Platform Ellipsoid Height</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 02 01 82 48 00 00 (CRC 27951)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Meters</td>
<td>uint16</td>
<td>2</td>
<td>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.</td>
</tr>
<tr>
<td>77</td>
<td>Operational Mode</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 03 21 00 00 00 (CRC 8938)</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>None</td>
<td>uint8</td>
<td>1</td>
<td>Indicates the mode of operations of the event portrayed in metadata. Enumerated. 0x00 = &quot;Other&quot; 0x01 = &quot;Operational&quot; 0x02 = &quot;Training&quot; 0x03 = &quot;Exercise&quot; 0x04 = &quot;Maintenance&quot;</td>
</tr>
</tbody>
</table>

23 October 2014 Motion Imagery Standards Board 23
### TAG  LS Name  Mapped US  ESD  ESD Name  US  US Name  Units  Format  Len  Notes

#### Frame Center Height Above Ellipsoid
- **TAG**: 78
- **LS Name**: Frame Center Height Above Ellipsoid
- **Mapped US**: 06 0E 2B 34 01 01 01 01 0E 01 02 03 48 00 00 00 (CRC 18095)
- **ESD**: x
- **ESD Name**: x
- **US**: x
- **US Name**: x
- **Units**: Meters
- **Format**: uint16
- **Len**: 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.

#### Sensor North Velocity
- **TAG**: 79
- **LS Name**: Sensor North Velocity
- **Mapped US**: 06 0E 2B 34 01 01 01 01 0E 01 02 02 7E 00 00 00 (CRC 59278)
- **ESD**: x
- **ESD Name**: x
- **US**: x
- **US Name**: x
- **Units**: Meters/Sec
- **Format**: int16
- **Len**: 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.

#### Sensor East Velocity
- **TAG**: 80
- **LS Name**: Sensor East Velocity
- **Mapped US**: 06 0E 2B 34 01 01 01 01 0E 01 02 02 7E 00 00 00 (CRC 37178)
- **ESD**: x
- **ESD Name**: x
- **US**: x
- **US Name**: x
- **Units**: Meters/Sec
- **Format**: int16
- **Len**: 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.

#### Image Horizon Pixel Pack
- **TAG**: 81
- **LS Name**: Image Horizon Pixel Pack
- **Mapped US**: 06 0E 2B 34 02 05 01 01 0E 01 03 02 08 00 00 00 (CRC 37658)
- **ESD**: x
- **ESD Name**: x
- **US**: x
- **US Name**: x
- **Units**: Pack
- **Format**: Pack
- **Len**: 5

- Corner Latitude Point 1 (Decimal Degrees)
  - 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.

- Corner Longitude Point 1 (Decimal Degrees)
  - 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.

- Corner Latitude Point 2 (Decimal Degrees)
  - 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.

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

23 October 2014  
Motion Imagery Standards Board  
25
<table>
<thead>
<tr>
<th>TAG</th>
<th>LS Name</th>
<th>Mapped US</th>
<th>ESD</th>
<th>ESD Name</th>
<th>US</th>
<th>US Name</th>
<th>Units</th>
<th>Format</th>
<th>Len</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>94</td>
<td>MIIS Core Identifier</td>
<td>Use ST1204 MIIS Core 16-byte Key.</td>
<td>x</td>
<td>x</td>
<td>06 0E 28 34 01 01 01 01 0E 01 04 05 03 00 00 00 (CRC 30280)</td>
<td>Motion Imagery Identification System Core</td>
<td>None</td>
<td>Binary Value</td>
<td>x</td>
<td>Local set tag to include the ST1204 MIIS Core Identifier Binary Value within ST0601. Use according to the rules and requirements defined in ST1204.</td>
</tr>
<tr>
<td>95</td>
<td>SAR Motion Imagery Metadata</td>
<td>Use ST1206 SARMI 16-byte Key.</td>
<td>x</td>
<td>x</td>
<td>06 0E 2B 34 02 08 01 01 0E 01 03 03 0D 00 00 00 (CRC 54900)</td>
<td>SAR Motion Imagery Metadata</td>
<td>None</td>
<td>Set</td>
<td>x</td>
<td>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.</td>
</tr>
</tbody>
</table>

Use -(2^31) as an "out of range" indicator.
-(2^31) = 0x80000000.
Res: ~42 nano deg.
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
   - Tag 13, Sensor Latitude
   - Tag 14, Sensor Longitude
   - 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.
   - Tag 5: Platform Heading Angle
   - Tag 6: Platform Pitch Angle
   - 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](image)

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Checksum</td>
<td></td>
<td>None</td>
<td>0..(2^16-1)</td>
<td>uint16</td>
</tr>
</tbody>
</table>

#### Notes
- 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.

#### Conversion Formula
- \( x \)
- \( x \)

#### Example Value
- Example LS Packet: \([K][L][V] = [0d][0d][0x8C ED]\)

#### Example 16-bit Checksum Code
```c
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 ()
```

#### 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

| LS Tag | 2  |
| LS Name | UNIX Time Stamp |
| US Mapped Key | Use EG0104 US Key |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microseconds</td>
<td>0..(2^64-1)</td>
<td>uint64</td>
</tr>
</tbody>
</table>

**Notes**
- 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.

**Conversion Formula**

**Example Value**
Oct. 24, 2008. 00:13:29.913

**Example LS Packet**
[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 05 00 00

**User Defined Time Stamp**
- 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.

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Mission ID</td>
<td>06 0E 2B 34 01 01 01 01</td>
<td>String</td>
<td>1..127</td>
<td>ISO 646</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0E 01 04 01 03 00 00 00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes
- Descriptive Mission Identifier to distinguish event or sortie.
- Value field is Free Text.
- Maximum 127 characters.

Conversion Formula
- x
- x

Example Value
MISSION01
[K][L][V] = [0d 3][0d 9][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

US Name
Episode Number
Mission Number

Notes
- x
- Number to distinguish different missions started on a given day.

To US:
- x

To ESD:
- x

To LS:
- x

US Conversion
ESD Conversion

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Platform Tail Number</td>
<td>06 0E 2B 34 01 01 01 01</td>
<td>String</td>
<td>1..127</td>
<td>ISO 646</td>
</tr>
<tr>
<td></td>
<td>0E 01 04 01 02 00 00 00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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

Conversion Formula

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

US Key | US Name | ESD Digraph | ESD Name | Platform Tail Number
× | × | | |

Notes
- ×

US Conversion | ESD Conversion
× | ×

To US:
- ×

To ESD:
- ×

To LS:
- ×

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Platform Heading Angle</td>
<td>Use EG0104 US Key</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>0..360</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula
- \( \text{LS}\_\text{dec} = \left( \frac{\text{LS}\_\text{range}}{\text{int}\_\text{range}} \right) \times \text{LS}\_\text{int} \) 
- \( \text{LS}\_5\_\text{dec} = \left( \frac{360}{65535} \right) \times \text{LS}\_5 \)

Example Value
159.9744 Degrees

Example LS Packet
06 0E 2B 34
01 01 01 07
07 01 10 01
06 00 00 00

US Key
06 0E 2B 34
01 01 01 07
07 01 10 01
06 00 00 00

ESD Digraph
Ih

ESD Name
UAV Heading (INS)

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.

US Conversion
- \( \text{US}\_\text{dec} = \left( \frac{360}{65535} \right) \times \text{LS}\_\text{uint} \)

ESD Conversion
- \( \text{ESD}\_\text{dec} = \left( \frac{360}{65535} \right) \times \text{LS}\_\text{uint} \)

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Platform Pitch Angle</td>
<td>Use EG0104 US Key</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>+/- 20</td>
<td>int16</td>
</tr>
</tbody>
</table>

**Notes**
- 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.

### Conversion Formula

\[ \text{LS}_{\text{dec}} = \left( \frac{\text{LS}_{\text{range}} \cdot \text{LS}_{\text{int}}}{\text{int}_{\text{range}}} \right) \]

\[ \text{LS}_{06}_{\text{dec}} = \left( \frac{40 \cdot 65534}{\text{LS}_{\text{int}}} \right) \]

### Example Value

-0.4315251 Degrees

### Example LS Packet

-06 0E 2B 34 01 01 01 07 07 01 10 01 05 00 00 00 0xFF 3D

#### US Key
-06 0E 2B 34 01 01 01 07

#### US Name
07 01 10 01 05 00 00 00 0x80

#### ESD Digraph
Ip

#### ESD Name
UAV Pitch (INS)

### US Conversion

\[ \text{US}_{\text{dec}} = \left( \frac{40 \cdot 65534}{\text{LS}_{\text{int}}} \right) \]

#### To US:
- US = (float)(40/0xFFFF * LS)

#### To LS:
- LS = (int16)round(0xFFFF/40 * US)

### ESD Conversion

\[ \text{ESD}_{\text{dec}} = \left( \frac{40 \cdot 65534}{\text{LS}_{\text{int}}} \right) \]

#### 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.
8.7 Tag 7: Platform Roll Angle Conversion

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Platform Roll Angle</td>
<td>Use EG0104 US Key</td>
<td>Degrees</td>
<td>+/- 50</td>
<td>int16</td>
</tr>
</tbody>
</table>

Notes
- Platform roll angle. Angle between transverse axis and transvers-longitudinal plane. Positive angles for lowered right wing.
- Map (-2^15-1) to (2^15-1) to +/-50.
- Use -(2^15) as "out of range" indicator.
- Res: ~1525 micro deg.
- Use -(2^15) as "out of range" indicator.
- (2^15) = 0x8000.

Conversion Formula
- LS_dec = \left( \frac{LS\_range}{int\_range} \right) \cdot LS\_int
- LS\_07\_dec = \left( \frac{100}{55534} \right) \cdot LS\_int

Example Value

<table>
<thead>
<tr>
<th>US Key</th>
<th>US Name</th>
<th>ESD Digraph</th>
<th>ESD Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 01 01 01 07</td>
<td>07 01 10 01 04 00 00 00</td>
<td>Ir</td>
<td>UAV Roll (INS)</td>
</tr>
</tbody>
</table>

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).

US Conversion
- US_dec = \left( \frac{100}{55534} \right) \cdot LS\_int

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.
8.8 Tag 8: Platform True Airspeed Conversion

| LS Tag | 8 |
| LS Name | Platform True Airspeed |
| US Mapped Key | 06 0E 2B 34 01 01 01 01 0E 01 01 0A 00 00 00 |
| Units | Range | Format |
| Meters/Second | 0..255 | uint8 |

Notes
- 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.

Conversion Formula
- LS_dec = LS_int
- LS_08_dec = round(LS_08)

Example Value
147 m/Sec

| US Key | x |
| US Name | x |
| ESD Digraph | As |
| ESD Name | True Airspeed |
| Units | Range | Format |
| x | x | x |
| Units | Range | Format |
| x | x | x |
| Notes |
- x

US Conversion
- To US:
  - x

ESD Conversion
- 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>Platform Indicated Airspeed</td>
<td>06 0E 2B 34 01 01 01 0E 01 01 0B 00 00 00</td>
<td>Meters/Second</td>
<td>0..255</td>
<td>uint8</td>
</tr>
</tbody>
</table>

**Notes**
- 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.

**Conversion Formula**

\[
\text{LS}_{\text{dec}} = \text{LS}_{\text{int}} \\
\text{LS}_{09}_{\text{dec}} = \text{round}(\text{LS}_{09})
\]

**Example Value**

159 m/Sec

**Example LS Packet**

<table>
<thead>
<tr>
<th>US Key</th>
<th>US Name</th>
<th>ESD Digraph</th>
<th>ESD Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>x</td>
<td>A1</td>
<td>Indicated Airspeed</td>
</tr>
</tbody>
</table>

**Units**
- Knots
  - Range: 0..999
  - Format: N

**Notes**
- Indicated airspeed of the aircraft.

**US Conversion**

\[
\text{ESD}_{\text{dec}} = \left(\text{LS}_{\text{uint}} \times \frac{1.94384449 \text{ knots}}{1 \text{ meters/second}}\right)
\]

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>Tag 10: Platform Designation</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td>Use EG0104 US Key</td>
</tr>
<tr>
<td>US Mapped Key</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>1..127</td>
<td>ISO 646</td>
</tr>
</tbody>
</table>

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.

Example Value
MQ1-B

<table>
<thead>
<tr>
<th>US Key</th>
<th>06 0E 2B 34 01 01 01 01</th>
</tr>
</thead>
<tbody>
<tr>
<td>US Name</td>
<td>01 01 20 01 00 00 00 00</td>
</tr>
<tr>
<td></td>
<td>Device Designation</td>
</tr>
</tbody>
</table>

ESD Digraph

Fc

ESD Name

Project ID Code

Notes
- Identifies the "house name" of the device used in capturing or generating the essence.
- 32 characters maximum.
- ISO7 character set.

US Conversion

<table>
<thead>
<tr>
<th>To US:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- x</td>
</tr>
</tbody>
</table>

ESD Conversion

<table>
<thead>
<tr>
<th>To ESD:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Convert string to Project ID Code.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>To LS:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- x</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>To ESD:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Convert Project ID Code to string.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>Image Source Sensor</td>
<td>Use EG0104 US Key</td>
<td>String</td>
<td>1..127</td>
<td>ISO 646</td>
</tr>
</tbody>
</table>

Notes
- 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.

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.

Example Value
<table>
<thead>
<tr>
<th>EO</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 01 01 01 01 04 20 01 02 01 01 00 00</td>
</tr>
</tbody>
</table>

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

US Key
- Indicates the type of the image source.
- 32 characters maximum.
- ISO7 character set.

Notes
- Identifies the source of the video image.
- 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

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>Image Coordinate System</td>
<td>Use EG0104 US Key</td>
<td>String</td>
<td>1..127</td>
<td>ISO 646</td>
</tr>
</tbody>
</table>

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

Conversion Formula
- To US:
  - x
- To LS:
  - x

Example Value

WGS-84

Example LS Packet

US Key

06 0E 2B 34
01 01 01 01

07 01 01 01
00 00 00 00

Image Coordinate System

ESD Digraph

Ic

ESD Name

Image Coordinate System

Notes
- Identifies the Digital Geographic Information Exchange Standard (DIGEST) georeferenced coordinate system used at image capture.
- ISO7 character set.

US Conversion
- x

ESD Conversion
- x

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 x 100km 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 |
| LS Name | Sensor Latitude |
| US Mapped Key | Use EGO0104 US Key |
| Notes | Conversion Formula |
| - Sensor Latitude. Based on WGS84 ellipsoid. | LS\_dec = \left( \frac{\text{LS\_range}}{\text{int\_range}} \right) \times \text{LS\_int} |
| - Map \(-2^{31}-1..2^{31}-1\) to +/-90. | LS\_13\_dec = \left( \frac{180}{4294967294} \right) \times \text{LS\_13} |
| - Use \(-2^{31}\) as an "error" indicator. | |
| - \(-2^{31} = 0x80000000.\) | |
| - Resolution: \(-42\) nano degrees. | |

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 |
| US Name | Device Latitude |
| ESD Digraph | Sa |
| ESD Name | Sensor Latitude |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>+/- 90</td>
<td>Double</td>
<td>Specifies a sensor's geographic location in decimal degrees of latitude.</td>
</tr>
<tr>
<td>Degrees</td>
<td>+/- 90.00</td>
<td>PDDMSST</td>
<td>Latitude of the aircraft. + Means North Latitude. All Latitude coordinates use WGS84.</td>
</tr>
</tbody>
</table>

**US Conversion**

\[
\text{US\_dec} = \left( \frac{180}{4294967294} \right) \times \text{LS\_int}
\]

**To US:**

- US = (double)(180/0xFFFFFFFFE * LS)

**To LS:**

- LS = (int32)round(0xFFFFFFFFE/180 * US)

**ESD Conversion**

\[
\text{ESD\_dec} = \left( \frac{180}{4294967294} \right) \times \text{LS\_int}
\]

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>Sensor Longitude</td>
<td>Use EG0104 US Key</td>
<td>Degrees</td>
<td>+/- 180</td>
<td>int32</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notes</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Sensor Longitude. Based on WGS84 ellipsoid.</td>
<td></td>
</tr>
<tr>
<td>- Map -(2^31-1)...(2^31-1) to +/-180.</td>
<td></td>
</tr>
<tr>
<td>- Use -(2^31) as an &quot;error&quot; indicator.</td>
<td></td>
</tr>
<tr>
<td>- -(2^31) = 0x80000000.</td>
<td></td>
</tr>
<tr>
<td>- Resolution: ~84 nano degrees.</td>
<td></td>
</tr>
</tbody>
</table>

**Example Value**

<table>
<thead>
<tr>
<th>128.426759042045 Degrees</th>
</tr>
</thead>
</table>

**Example LS Packet**

<table>
<thead>
<tr>
<th>US Key</th>
<th>ESD Digraph</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 01 01 01 03</td>
<td>So</td>
</tr>
<tr>
<td>07 01 02 01 02 06 02 00</td>
<td>Sensor Longitude</td>
</tr>
</tbody>
</table>

**Conversion Formula**

\[
\text{LS}_{\text{dec}} = \frac{360}{4294967294} \times \text{LS}_{\text{int}}
\]

**US Conversion**

\[
\text{US}_{\text{dec}} = \frac{360}{4294967294} \times \text{LS}_{\text{int}}
\]

**ESD Conversion**

\[
\text{ESD}_{\text{dec}} = \frac{360}{4294967294} \times \text{LS}_{\text{int}}
\]

**Notes**

- Longitude of the aircraft. + Means East Longitude. All Longitude coordinates use WGS84.
- Longitude of the aircraft. + Means East Longitude. All Longitude coordinates use WGS84.

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>Sensor True Altitude</td>
<td>Use EGO104 US Key</td>
<td>Meters</td>
<td>-900..19000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes:
- 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.

Conversion Formula:

\[
LS_{\text{dec}} = \left( \frac{\text{uint\_range}}{65535} \times \text{LS\_uint} \right) - \text{Offset}
\]

\[
LS_{15\_\text{dec}} = \left( \frac{19900}{65535} \times \text{LS\_15} \right) - 900
\]

Example Value

- 14190.72 Meters

Example LS Packet

[06 0E 2B 34 01 01 01 01 07 01 02 01 02 02 00 00 0x0] = [0d15][0d2][0xC2 21] = [K][L][V] = [K][L][N]

US Key

- 06 0E 2B 34 01 01 01 01
- Device Altitude

ESD Digraph

- SL

ESD Name

- Sensor Altitude

Units | Range | Format |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Meters</td>
<td>Float</td>
<td>Float</td>
</tr>
<tr>
<td>Feet</td>
<td>+/- 0..99,999</td>
<td>FN</td>
</tr>
</tbody>
</table>

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

US Conversion

\[
\text{US\_dec} = \left( \frac{19900}{65535} \times \text{LS\_uint} \right) - 900
\]

To US:
- \( \text{US} = \text{float}\left(\left(\frac{19900}{\text{0xFFFF}}\right) \times \text{LS} - 900\right) \)

To LS:
- \( \text{LS} = \text{uint16}\left(\text{0xFFFF/19900} \times (\text{US} + 900)\right) \)

ESD Conversion

\[
\text{ESD\_dec} = \left( \frac{19900}{65535} \times \text{LS\_uint} - 900 \right) \times 3.2808399\text{ft} \frac{1m}{1}\text{m}
\]

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 |
| US Mapped Key | Sensor Horizontal field of View |
| Use EG0104 US Key |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>0..180</td>
<td>uint16</td>
</tr>
</tbody>
</table>

| 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 = (LS_range / uint_range) * LS_uint |
| LS_16_dec = (180 / 65535) * LS_16 |

| Example Value |
| 144.5713 Degrees |

| Example LS Packet |
| 06 0E 2B 34 01 01 02 04 02 01 01 08 00 00 |

| US Key |
| 06 0E 2B 34 01 01 02 |

| US Name |
| 04 02 01 01 08 00 00 |

| ESD Digraph |
| Fv |

| ESD Name |
| field of View |

| Field of View (FOV-Horizontal) |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>0..180</td>
<td>Float</td>
</tr>
</tbody>
</table>

| Notes |
| Sensor Horizontal field of view. |

| US Conversion |
| US_dec = (180 / 65535) * LS_uint |

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

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

| ESD Conversion |
| ESD_dec = (180 / 65535) * LS_uint |

| 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>Sensor Vertical Field of View</td>
<td>06 0E 2B 34 01 01 01 07 04 02 01 01 0A 01 00</td>
<td>Degrees</td>
<td>0..180</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula

\[
\text{LS}_\text{dec} = \left( \frac{\text{LS}_\text{range}}{\text{uint}_\text{range}} \right) \times \text{LS}_\text{uint}
\]

\[
\text{LS}_{17}\_\text{dec} = \left( \frac{180}{65535} \right) \times \text{LS}_{17}
\]

Example Value

152.6436 Degrees

Example LS Packet

[K][L][V] = [0d17][0d2][0xD9 17]

US Name: Vertical Field of View

Notes
- 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

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

ESD Conversion

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

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>Sensor Relative Azimuth Angle</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 02 04 00 00 00</td>
</tr>
</tbody>
</table>

**Units** | **Range** | **Format**
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>0..360</td>
<td>uint32</td>
</tr>
</tbody>
</table>

**Notes**
- 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.

**Conversion Formula**
\[
LS_{\text{dec}} = \left( \frac{\text{uint}_{\text{range}} \times \text{LS}_{\text{uint}}}{4294967295} \right)
\]
\[
LS_{18\text{dec}} = \left( \frac{360}{4294967295} \times \text{LS}_{18} \right)
\]

**Example Value**
160.719211474396 Degrees

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

**US**

- **Key**
- **Name**

**Units** | **Range** | **Format**
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>0..359.99</td>
<td>DDD.HH</td>
</tr>
</tbody>
</table>

**Notes**
- 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.

**Conversion**

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

Notes
- Relative Elevation Angle of sensor to platform longitudinal-transverse plane. Negative angles down.
- Map -2^31..(2^31-1) to +/-180.
- Use -2^31 as an "error" indicator.
- Res: ~84 ndeg.

Example Value
-168.79234833941 Degrees

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

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

US Conversion

| To US: |
| - x |

ESD Conversion

| 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>Sensor Relative Roll Angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td>Local Set</td>
</tr>
<tr>
<td>US Mapped Key</td>
<td>06 0E 2B 34 01 01 01 06 00 00 00</td>
</tr>
<tr>
<td>Units</td>
<td>Range</td>
</tr>
<tr>
<td>Degrees</td>
<td>0..360</td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula
\[
\text{LS} = \left( \frac{\text{LS range}}{\text{uint range}} \right) \times \text{LS uint}
\]
\[
\text{LS}_{20} = \left( \frac{360}{2^{32} - 1} \right) \times \text{LS}_{20}
\]

Example Value
176.865437690572 Degrees

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

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>Slant Range</td>
<td>Use EG0104 US Key</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meters</td>
<td>0..5,000,000</td>
<td>uint32</td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula
\[
\text{LS}_\text{dec} = \frac{\text{LS}_\text{range}}{\text{uint}_\text{range}} \times \text{LS}_\text{uint}
\]
\[
\text{US}_\text{dec} = \frac{5000000}{4294967295} \times \text{LS}_\text{uint}
\]

Example Value
68590.98 Meters

Example LS Packet
[K][L][V] = [0x21][0x4d][0x03 83 09 26]

Example US Packet
06 0E 2B 34 01 01 01 01
07 01 08 01 01 00 00 00

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
\[
\text{US}_\text{dec} = \left( \frac{5000000}{4294967295} \right) \times \text{LS}_\text{uint}
\]

To US:
- US = (float)(5000000/0xffffffff * LS)

To LS:
- LS = (uint32)round(0xffffffff/5000000 * US)

ESD Conversion
\[
\text{ESD}_\text{dec} = \frac{5000000}{1852} \times \text{LS}_\text{uint}
\]

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.
As of ST 0601.3 Generic Flag Data 01 (Tag 47) contains a flag which indicates whether 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 |
| LS Name | Target Width |
| US Mapped Key | Use EG0104 US Key |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meters</td>
<td>0..10,000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes
- Target Width within sensor field of view.
- Map 0..(2^16-1) to 0..10000 meters.
- 1 meter = 3.2808399 feet.
- Resolution: ~.16 meters.

Conversion Formula
- $\text{LS}_{\text{dec}} = \left( \frac{\text{LS}_{\text{range}}}{\text{uint\_range}} \right) \times \text{LS}_{\text{uint}}$
- $\text{LS}_{\text{dec}} = \left( \frac{10000}{65535} \right) \times \text{LS}_{\text{uint}}$

Example Value
- 722.8199 Meters

Example LS Packet
- 06 0E 2B 34 01 01 01 01
- 07 01 09 02 01 00 00 00

US Key
- Target Width

US Name
- Target Width

ESD Digraph
- Target Width

ESD Name
- Target Width

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
- $\text{US}_{\text{dec}} = \left( \frac{10000}{65535} \right) \times \text{LS}_{\text{uint}}$

To US:
- US = \((\text{float})(10000/0xFFFF \times \text{LS})\)

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](image-url)
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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>Frame Center Latitude</td>
<td>Use EG0104 US Key</td>
<td>Degrees</td>
<td>+/- 90</td>
<td>int32</td>
</tr>
</tbody>
</table>

**Notes**
- 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.

**Conversion Formula**

\[
\text{LS}_\text{dec} = \left( \frac{\text{LS}_\text{range}}{2^{31} - 1} \right) \times \text{LS}_\text{int}
\]

\[
\text{LS}_{23}\_\text{dec} = \left( \frac{180}{4294967294} \right) \times \text{LS}_23
\]

**Example Value**

-10.542386331461 Degrees

**Example LS Packet**

\[K][L][V] = \{0d23\}[Od4]\{0xF1 01 A2 29\}

**US Key**

06 0E 2B 34 01 01 01 01
07 01 02 01 03 02 00 00
Frame Center Latitude

**ESD Digraph**

Ta

**ESD Name**

Target Latitude

**Notes**
- Specifies the video frame center point geographic location in decimal degrees of latitude.
- Positive values indicate northern hemisphere.
- Negative values indicate southern hemisphere.

**US Conversion**

\[
\text{US}_\text{dec} = \left( \frac{180}{4294967294} \right) \times \text{LS}_\text{int}
\]

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

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

**ESD Conversion**

\[
\text{ESD}_\text{dec} = \left( \frac{180}{4294967294} \right) \times \text{LS}_\text{int}
\]

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
<td>Frame Center Longitude</td>
<td>Degrees</td>
<td>+/- 180</td>
<td>int32</td>
</tr>
</tbody>
</table>

### Notes
- 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.

### Conversion Formula
- \[ \text{LS}_{\text{dec}} = \left( \frac{\text{int}_\text{range} \times \text{LS}_{\text{int}}}{2^{31}} \right) \]
- \[ \text{LS}_{24\text{dec}} = \left( \frac{360}{4294967294} \times \text{LS}_{24} \right) \]

### Example Value
29.157890122923 Degrees

### LS Packet Example
[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

### ESD Name
Target Longitude

### Notes
- Specifies the video frame center point geographic location in decimal degrees of longitude.
- Positive values indicate eastern hemisphere.
- Negative values indicate western hemisphere.

#### US Conversion
- \[ \text{US}_{\text{dec}} = \left( \frac{360}{4294967294} \times \text{LS}_{\text{int}} \right) \]
- \[ \text{US} = \text{double}(360/0xFFFFFFFF * \text{LS}) \]

#### ESD Conversion
- \[ \text{ESD}_{\text{dec}} = \left( \frac{360}{4294967294} \times \text{LS}_{\text{int}} \right) \]
- \[ \text{ESD} = \text{double}(360/0xFFFFFFFF * \text{LS}) \]

### Notes
- Longitude of the EO/IR payload's aimpoint on the ground. + Means East longitude. All longitude coordinates use WGS84.

### 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             |
| LS Name | Frame Center Elevation |
| US Mapped Key | 06 0E 2B 34 01 01 01 0A 07 01 02 01 03 16 00 00 |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meters</td>
<td>-900..19000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes:
- Terrain elevation at frame center relative to Mean Sea Level (MSL).
- Map 0..(2^16-1) to -900..19000 meters.
- Resolution: ~0.3 meters.

Conversion Formula:
- \( \text{LS}_{\text{dec}} = \left( \frac{\text{uint}_{\text{range}} + \text{LS}_{\text{uint}}}{\text{Offset}} \right) \)
- \( \text{LS}_{25}_{\text{dec}} = \left( \frac{19900 \times \text{LS}_{25}}{65535} \right) - 900 \)

Example Value

| LS Packet
|-----------------
| [k][l][v] = [0d25][0d2][0x34F3] |

Example LS Packet

| US Key | x |
| US Name | x |

| ESD Digraph | Frame Center Elevation |
| ESD Name | |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feet</td>
<td>+/- 0..99,999</td>
<td>PN</td>
</tr>
</tbody>
</table>

Notes:
- Terrain elevation at frame center.

US Conversion

| x |

<table>
<thead>
<tr>
<th>ESD Conversion</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{ESD}<em>{\text{dec}} = \left( \frac{19900 \times \text{LS}</em>{\text{uint}}-900}{65535} \right) \times 3.2808399ft/\text{m} )</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>Offset Corner Latitude Point 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>US Mapped Key</td>
<td>Use EG0104 US Key</td>
</tr>
</tbody>
</table>

#### Notes

- Frame Latitude, offset for upper left corner. Based on WGS84 ellipsoid.
- Use with Frame Center Latitude.
- Map -(2^15-1) to +/-0.075.
- Use -(2^15) as an "error" indicator.
- Resolution: ~1.2micro deg, ~0.25meters at equator.

#### Conversion Formula

- **To US:**
  - US = (double)((0.15/0xFFFE * LS) + LS_23_dec)

- **To LS:**
  - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LAT))

#### Example LS Packet

```
-10.57963799987 Corrected Degrees
[K][L][V] = [0d26][0d1][0xC0 6E]
```

#### Example ESD Conversion

```
US dec = (0.15/65534 * LS_int) + LS_23_dec
ESD dec = (0.15/65534 * LS_int) + LS_23_dec
```

#### Example Value

```
Example Value
-10.57963799987 Corrected Degrees
```

### 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td>Offset Corner Longitude Point 1</td>
<td>Degrees</td>
<td>+/-0.075</td>
<td>int16</td>
</tr>
</tbody>
</table>

**Notes**
- 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.
- Resolution: ~1.2micro deg, ~0.25meters at equator.

**Conversion Formula**
- LS\(_{\text{dec}}\) = \((\text{LS range}_{\text{int}} \times \text{LS\(_{\text{int}}\)}) + \text{LS\(_{24}\)_dec}\)
- LS\(_{27}\)_dec = \((0.15 / \text{LS\(_{27}\)}) + \text{LS\(_{24}\)_dec}\)

**Example Value**
- Example LS Packet: 29.1237677986333 Corrected
  
  **US Key (Decimlar Degrees)**
  
  06 0E 2B 34 01 01 01 03
  07 01 02 01 03 08 01 00
  Corner Longitude Point 1
  (Decimal Degrees)

  **ESD Digraph**
  
  SAR Longitude 4

  **US Conversion**
  
  \(\text{US}_\text{dec} = (0.15 / \text{LS}_{\text{int}}) + \text{LS}_{24}\)_dec

  **ESD Conversion**
  
  \(\text{ESD}_\text{dec} = (0.15 / \text{LS}_{\text{int}}) + \text{LS}_{24}\)_dec

  **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.

**Example LS Packet**

[K][L][V] = [0d27][0d2][0xCB E9]

**Notes**
- Longitude coordinate of corner 1 of an image or bounding rectangle.
- Positive (+) is eastern hemisphere.
- Negative (-) is western hemisphere.

**US Conversion**

\(\text{US}_\text{dec} = (0.15 / \text{LS}_{\text{int}}) + \text{LS}_{24}\)_dec

**ESD Conversion**

\(\text{ESD}_\text{dec} = (0.15 / \text{LS}_{\text{int}}) + \text{LS}_{24}\)_dec

**Notes**
- Convert LS to decimal.
- Convert decimal to ASCII.
- 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>Offset Corner Latitude Point 2</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td>Use EG0104 US Key</td>
<td>Degrees</td>
<td>+/- 0.075</td>
<td>int16</td>
</tr>
</tbody>
</table>

Notes
- 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.
- Resolution: ~1.2 micro deg, ~0.25 meters at equator.

Conversion Formula
- \( \text{LS}_{\text{dec}} = \frac{\text{LS range}}{\text{int range}} \times \text{LS}_{\text{int}} + \text{LS}_{23\text{ dec}} \)
- \( \text{LS}_{28\text{ dec}} = \frac{0.15}{65534} \times \text{LS}_{28} + \text{LS}_{23\text{ dec}} \)

Example Value
-10.5661816260963 Corrected Degrees

Example LS Packet
[K][L][V] = [0d28][0d2][0xD7 65]

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).
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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>Offset Corner Longitude Point 2</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td>Use EG0104 US Key</td>
<td>Degrees</td>
<td>+/- 0.075</td>
<td>int16</td>
</tr>
</tbody>
</table>

**Notes**
- 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.
- Resolution: ~1.2 micro deg, ~0.25 meters at equator.

**Conversion Formula**
\[
\text{LS}_{\text{dec}} = \left(\frac{\text{LS}_{\text{range}}}{\text{int\_range}}\right) \text{LS}_{\text{int}} + \text{LS}_{\text{24\_dec}}
\]
\[
\text{LS}_{\text{29\_dec}} = \left(\frac{0.15}{65534}\right) \text{LS}_{\text{29}} + \text{LS}_{\text{24\_dec}}
\]

**Example Value**
Corner Longitude Point 2

**Example LS Packet**
29.140824172424 Corrected

**US Key**
06 0E 2B 34 01 01 01 03
07 01 02 01 03 0C 01 00
Corner Longitude Point 2

**ESD Name**
SAR Longitude 1

**US Conversion**
\[
\text{US}_{\text{dec}} = \left(\frac{0.15}{65534}\right) \text{LS}_{\text{int}} + \text{LS}_{\text{24\_dec}}
\]

**To US:**
- US = (double)((0.15/0xFFFE * LS) + LS_24_dec)

**To LS:**
- LS = (int16)round(0xFFFE/0.15 * (US - Frame\_Center\_LON))

**ESD Conversion**
\[
\text{ESD}_{\text{dec}} = \left(\frac{0.15}{65534}\right) \text{LS}_{\text{int}} + \text{LS}_{\text{24\_dec}}
\]

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>Offset Corner Latitude Point 3</td>
<td>Use EG0104 US Key</td>
<td>Degrees</td>
<td>+/-0.075</td>
<td>int16</td>
</tr>
</tbody>
</table>

Notes:
- 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.
- Resolution: ~1.2micro deg, ~0.25meters at equator.

Conversion Formula:
- \( \text{LS}_30\text{dec} = \left( \frac{0.15}{65534} \times \text{LS}_\text{int} \right) + \text{LS}_23\text{dec} \)
- \( \text{LS}_\text{dec} = \left( \frac{\text{LS}_\text{range}}{\text{int}_\text{range}} \times \text{LS}_\text{int} \right) + \text{LS}_23\text{dec} \)

Example Value
- Conversion Formula

Example LS Packet
- Tag 30: Offset Corner Latitude Point 3

Example US Packet
- 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.
- Resolution: ~1.2micro deg, ~0.25meters at equator.

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 (see Figure 8-14).
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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>Offset Corner Longitude Point 3</td>
<td>Degrees</td>
<td>+/-0.075</td>
<td>int16</td>
</tr>
</tbody>
</table>

Use EG0104 US Key

Notes
- 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.

Conversion Formula

\[ \text{LS}_\text{dec} = \left( \frac{\text{LS}_\text{range}}{\text{int}_\text{range}} \right) \text{LS}_\text{int} + \text{LS}_\text{24}_\text{dec} \]

\[ \text{LS}_31\_\text{dec} = \left( \frac{0.15}{0xFFFF} \right) \text{LS}_\text{31} + \text{LS}_\text{24}_\text{dec} \]

Example Value

29.1542782573265 Corrected [K][L][V] = [0d31][0d2][0xF9 D6]

Example LS Packet

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

US Name

ESD Digraph
SAR Longitude 2

ESD Name

Notes
- Longitude coordinate of corner 3 of an image or bounding rectangle.
- Positive (+) is eastern hemisphere.
- Negative (-) is western hemisphere.

US Conversion

\[ \text{US}_\text{dec} = \left( \frac{0.15}{0xFFFF} \right) \text{LS}_\text{int} + \text{LS}_\text{24}_\text{dec} \]

To US:
- US = (double)((0.15/0xFFFE * LS) + LS_24_dec)

ESD Conversion

\[ \text{ESD}_\text{dec} = \left( \frac{0.15}{0xFFFF} \right) \text{LS}_\text{int} + \text{LS}_\text{24}_\text{dec} \]

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>Offset Corner Latitude Point 4</td>
<td>Degrees</td>
<td>+/-0.075</td>
<td>int16</td>
</tr>
</tbody>
</table>

**Notes**
- Frame Latitude, offset for lower left corner. Based on WGS84 ellipsoid.
- Use with Frame Center Latitude.
- Map -2^15..-2^15 to +/-0.075.
- Use -2^15 as an "error" indicator.
- Resolution: ~1.2micro deg, ~0.25meters at equator.

**Conversion Formula**

\[
\begin{align*}
\text{LS}_\text{dec} &= \left( \frac{\text{LS}_\text{range}}{\text{int}_\text{range}} \right) \text{LS}_\text{int} + \text{LS}_\text{23}_\text{dec} \\
\text{LS}_32\_\text{dec} &= \left( \frac{0.15}{65534} \right) \text{LS}_32 + \text{LS}_\text{23}_\text{dec}
\end{align*}
\]

**Example Value**

-10.5392711674031 Corrected Degrees

**Example LS Packet**

[K][L][V] = [0d32][0d42][0x05 0x52]

**US Conversion**

\[
\begin{align*}
\text{US}_\text{dec} &= \left( \frac{0.15}{65534} \right) \text{LS}_\text{int} + \text{LS}_\text{23}_\text{dec} \\
\text{To US:} & \quad \text{US} = (\text{double})(0.15/0xFFFE * \text{LS}) + \text{LS}_\text{23}_\text{dec} \\
\text{To LS:} & \quad \text{LS} = (\text{int16})\text{round}(0xFFFE/0.15 * (\text{US} - \text{Frame Center LAT}))
\end{align*}
\]

**ESD Conversion**

\[
\begin{align*}
\text{ESD}_\text{dec} &= \left( \frac{0.15}{65534} \right) \text{LS}_\text{int} + \text{LS}_\text{23}_\text{dec} \\
\text{To ESD:} & \quad \text{ESD} = (\text{double})(0.15/0xFFFE * \text{LS}) + \text{LS}_\text{23}_\text{dec} \\
\text{To LS:} & \quad \text{LS} = (\text{int16})\text{round}(0xFFFE/0.15 * (\text{ESD} - \text{Frame Center LAT}))
\end{align*}
\]

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).
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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>Offset Corner Longitude Point 4</td>
<td>Degrees</td>
<td>+/- 0.075</td>
<td>int16</td>
</tr>
</tbody>
</table>

**Notes**
- Frame Longitude, offset for lower left corner. Based on WGS84 ellipsoid.
- Use with Frame Center Longitude.
- Map -2^15-1 to 2^15-1 to +/-0.075.
- Use -2^15 as an "error" indicator.
- Resolution: ~1.2micro deg, ~0.25meters at equator.

**Conversion Formula**
- LS_dec = \( \frac{LS\_range}{int\_range} \times LS\_int \) + LS_24_dec
- LS_33_dec = \( 0.15 \times \frac{LS\_33}{65534} \) + LS_24_dec

**Example Value**
29.1677346311172 Corrected
KLV = [0d33][0d2][0x10 CD]

**US Conversion**
- US_dec = \( \frac{0.15}{65534} \times LS\_int \) + LS_24_dec
- To US:
  - US = (double)((0.15/0xFFFE * LS) + LS_24_dec)
  - US_24_dec = int16(round(0xFFFE/0.15 * (US - Frame_Center_LON)))
- To LS:
  - LS = (int16)round(0xFFFE/0.15 * (US - Frame_Center_LON))

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 |
| LS Name | Icing Detected |
| US Mapped Key | 06 0E 2B 34 01 01 01 01 0E 01 01 0C 00 00 00 |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Icing Code</td>
<td>0..255</td>
<td>uint8</td>
</tr>
</tbody>
</table>

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

Conversion Formula

Example Value

Example LS Packet

Invalid Icing Code

| US Key | x |
| US Name | x |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Icing Code</td>
<td>0..2</td>
<td>N</td>
</tr>
</tbody>
</table>

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

US Conversion

| ESD Conversion |
| To US: | x |
| To ESD: | - Convert string to ID code. |
| 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 |
| LS Name | Wind Direction |
| US Mapped Key | 06 0E 2B 34 01 01 01 0E 01 01 0D 00 00 00 |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>0..360</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula
- LS_dec = \( \frac{LS\_range}{uint\_range} \) \times LS\_uint 
- LS\_35\_dec = \( \frac{360}{65535} \) \times LS\_35

Example Value
- 235.924 Degrees

Example LS Packet
- [K][L][V] = [0d35][0d2][0xA7 C4]

US Conversion
- To US:
- To LS:

ESD Conversion
- 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  |
| LS Name | Wind Speed |
| US Mapped Key | 06 0E 2B 34 01 01 01 0E 01 01 0E 00 00 00 |

**Units** | **Range** | **Format**
---|---|---
Meters/Second | 0..100 | uint8

**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{\text{uint_range}}{255}\right) \times \text{LS_uint}
```

```
LS_36_dec = \left(\frac{100}{255}\right) \times \text{LS_36}
```

**Example Value**

| US Key | 69.80392 m/s |
| US Mapped Key | | |

**Example LS Packet**

```
[0x0] [0x1] [0x2] = [0xA2] [0x01] [0x01]
```

**US Conversion**

- **To US:**
  - ESD dec = \left(\frac{100}{255}\right) \times \text{LS_uint} \times 1.94384449 \text{knots/m/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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>37</td>
<td>Static Pressure</td>
<td>06 0E 2B 34 01 01 01 0E 01 01 01 0F 00 00 00</td>
<td>Millibar</td>
<td>0..5000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes
- Static pressure at aircraft location.
- Map 0..(2^16-1) to 0..5000 mbar.
- 1 mbar = 0.0145037738 PSI.
- Resolution: ~0.08 Millibar

Conversion Formula
\[ \text{LS}_{\text{dec}} = \left( \text{LS}_{\text{range}} \times \text{LS}_{\text{uint}} \right) \]
\[ \text{LS}_{37}_{\text{dec}} = \left( \frac{5000}{65535} \times \text{LS}_{37} \right) \]

Example Value
3725.185 mbar

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

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>Density Altitude</td>
<td>Meters</td>
<td>-900..19000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notes</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Density altitude at aircraft location.</td>
<td>LS_dec = (LS_range * LS_uint) - Offset</td>
</tr>
<tr>
<td>Relative aircraft performance metric</td>
<td>LS_38_dec = (19900 * LS_38) - 900</td>
</tr>
<tr>
<td>based on outside air temperature, static</td>
<td></td>
</tr>
<tr>
<td>pressure, and humidity.</td>
<td></td>
</tr>
<tr>
<td>- Map 0..(2^16-1) to -900..19000 meters.</td>
<td></td>
</tr>
<tr>
<td>- Offset = -900.</td>
<td></td>
</tr>
<tr>
<td>- 1 meter = 3.2808399 feet.</td>
<td></td>
</tr>
<tr>
<td>- Resolution: ~0.3 meters.</td>
<td></td>
</tr>
</tbody>
</table>

**Example Value**

<table>
<thead>
<tr>
<th>14818.68 Meters</th>
</tr>
</thead>
<tbody>
<tr>
<td>[K][L][V] = [0d38][0d2][0xCA 35]</td>
</tr>
</tbody>
</table>

**Example LS Packet**

<table>
<thead>
<tr>
<th>US Key</th>
<th>US Name</th>
<th>ESD Digraph</th>
<th>ESD Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>x</td>
<td>Da</td>
<td>Density Altitude</td>
</tr>
</tbody>
</table>

**US Conversion**

<table>
<thead>
<tr>
<th>Notes</th>
<th>ESD Conversion</th>
</tr>
</thead>
<tbody>
<tr>
<td>- x</td>
<td>ESD_dec = (19900 * LS_uint-900) / 3.2808399 ft</td>
</tr>
<tr>
<td>- Density Altitude of the</td>
<td></td>
</tr>
<tr>
<td>aircraft.</td>
<td></td>
</tr>
</tbody>
</table>

**To US:**

- x

**To LS:**

- x

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

| LS Tag | 39 |
| LS Name | Outside Air Temperature |
| US Mapped Key | 06 0E 2B 34 01 01 01 01 0E 01 01 01 11 00 00 00 |

### Units, Range, Format

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Celcius</td>
<td>-128..+127</td>
<td>int8</td>
</tr>
</tbody>
</table>

### Notes

- Temperature outside of aircraft.
- -128..127 Degrees Celsius.
- Resolution: 1 degree celsius.

### Conversion Formula

\[ \text{LS}_{\text{dec}} = \text{LS}_{\text{int}} \]
\[ \text{LS}_{39\_\text{dec}} = \text{LS}_{39} \]

### Example Value

- **Example Value**

  84 Celcius

- **Example LS Packet**

  \[ [K][L][V] = [0d39][0d1][0x54] \]

### US Name

- **US Name**

  x

### ESD Digraph

- **ESD Name**

  At

### US Conversion

- **Units**

  x

- **Range**

  x

- **Format**

  x

### ESD Conversion

- **Units**

  Celcius

- **Range**

  +/- 99

- **Format**

  PDD

### Notes

- Outside air temperature measured at the aircraft

### To US:

- Convert int8 to string.

### To LS:

- Convert string to int8.

### Notes

- Outside air temperature measured at the aircraft

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>Target Location Latitude</td>
<td>Degrees</td>
<td>+/- 90</td>
<td>int32</td>
<td></td>
</tr>
<tr>
<td></td>
<td>06 0E 2B 34 01 01 01 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0E 01 01 03 02 00 00 00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes:
- 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.

Conversion Formula:

\[
\text{LS}_{\text{dec}} = \left( \frac{\text{LS}_{\text{range}} \times \text{LS}_{\text{int}}}{\text{int}_{\text{range}}} \right)
\]

\[
\text{LS}_{40\text{dec}} = \left( \frac{180}{2^{31}-1} \right) \times \text{LS}_{40}
\]

Example Value

-79.1638500518929 Degrees

Example LS Packet

[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

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

Notes
- 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.

Conversion Formula

\[
\text{LS}_{\text{dec}} = \left( \text{LS}_{\text{range}} \times \text{LS}_{\text{int}} \right)
\]

\[
\text{LS}_{41\text{dec}} = \left( \frac{360}{21996447296} \right) \times \text{LS}_{41}
\]

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>Target Location Elevation</td>
<td>Meters</td>
<td>-900..19000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notes</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Calculated target elevation. This is the crosshair location if different from frame center.</td>
<td>LS_dec = ( \frac{\text{LS range}_{\text{uint16}} \times \text{LS_uint}}{\text{Offset}} )</td>
</tr>
<tr>
<td>- Map 0..(2^16-1) to -900..19000 meters.</td>
<td>LS_42_dec = ( \frac{19900 \times \text{LS}_42}{65535} - 900 )</td>
</tr>
<tr>
<td>- Offset = -900.</td>
<td></td>
</tr>
<tr>
<td>- 1 meter = 3.2808399 feet.</td>
<td></td>
</tr>
<tr>
<td>- Resolution: ~0.3 meters.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>18389.05 Meters</td>
<td>[K][L][V] = {0d42}[0d2][0xF8 23]</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>43</td>
<td>Target Track Gate Width</td>
<td></td>
</tr>
<tr>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 03 05 00 00 00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pixels</td>
<td>0..512</td>
<td>uint8</td>
</tr>
</tbody>
</table>

Notes
- Tracking gate width (x value) of tracked target within field of view.
- Closely tied to source video resolution in pixels.

Conversion Formula
- \( LS_{dec} = 2 \times LS_{uint} \)
- \( LS_{43\_dec} = \text{round}(2 \times LS_{43}) \)

Example Value
- 6 Pixels

Example LS Packet
- \([K][L][V] = [0d43][0d1][0x03]\)

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>44</td>
<td>Target Track Gate Height</td>
<td>Pixels</td>
<td>0..512</td>
<td>uint8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notes</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Tracking gate height (y value) of tracked target within field of view.</td>
<td></td>
</tr>
<tr>
<td>- Closely tied to source video resolution in pixels.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LS_dec = 2 * LS_uint</td>
</tr>
<tr>
<td></td>
<td>LS_44_dec = round(2 * LS_44)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>30 Pixels</td>
<td>[K][L][V] = [0][d44][0d1][0x0F]</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>45</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td>Target Error Estimate - CE90</td>
</tr>
<tr>
<td>US Mapped Key</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 03 07 00 00 00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meters</td>
<td>0..4095</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes
- 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

Conversion Formula
- \( LS_{\text{dec}} = \left( \text{LS\_range} \times \text{LS\_uint} \right) \)
- \( LS_{45\_\text{dec}} = \left( \frac{4095}{65535} \times \text{LS\_45} \right) \)

Example Value: Example LS Packet
425.319 Meters \([K][L][V] = [0d 45][0d 2][0x1A 95] \)

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:
\[
Q = \begin{bmatrix}
\sigma_x^2 & \sigma_{xn} & \sigma_{en} \\
\sigma_{nx} & \sigma_n^2 & \sigma_{na} \\
\sigma_{xn} & \sigma_{an} & \sigma_n^2
\end{bmatrix}
\]

Min and Max Sigma Values:
\[
\sigma_{\text{max}}^2 = \frac{\left(\sigma_x^2 + \sigma_n^2\right) + \sqrt{\left(\sigma_x^2 + \sigma_n^2\right)^2 - 4\left(\sigma_x^2\sigma_n^2 - \sigma_{en}^2\right)}}{2}
\]
\[
\sigma_{\text{min}}^2 = \frac{\left(\sigma_x^2 + \sigma_n^2\right) - \sqrt{\left(\sigma_x^2 + \sigma_n^2\right)^2 - 4\left(\sigma_x^2\sigma_n^2 - \sigma_{en}^2\right)}}{2}
\]

CE90 represents the 90 percent probability circular error radius of absolute horizontal accuracy. With \( \sigma_{\text{max}} \) and \( \sigma_{\text{min}} \) known, the Circular Error for 90% confidence can be calculated as:
\[
CE90 = \sigma_{\text{max}} \cdot a\left(\frac{\sigma_{\text{min}}}{\sigma_{\text{max}}}\right) \quad \text{where} \quad a(x) = 0.4194x^2 + 0.0774x + 1.648.
\]

This is one means for determining CE90 from statistical data in the easting-northing-up coordinate system, yet similar calculations are allowed.

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

---

23 October 2014  Motion Imagery Standards Board  94
8.46 Tag 46: Target Error Estimate - LE90 Conversion

| LS Tag | 46 |
| LS Name | Target Error Estimate - LE90 |
| US Mapped Key | 06 0E 2B 34 01 01 01 01 0E 01 01 03 08 00 00 00 |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meters</td>
<td>0..4095</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes
- 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

Conversion Formula
- \( LS_{\text{dec}} = \left( \frac{\text{LS range}}{\text{uint range}} \right) \times \text{LS\_uint} \)
- \( LS_{46\_\text{dec}} = \frac{4095}{65535} \times \text{LS\_46} \)

Example Value
- 609.0718 Meters
- Example LS Packet: [K][L][V] = [0d 46][0d 2][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:

\[
Q = \begin{bmatrix}
\sigma_v^2 & \sigma_{vn} & \sigma_{vu} \\
\sigma_{nv} & \sigma_n^2 & \sigma_{nu} \\
\sigma_{uv} & \sigma_{nu} & \sigma_u^2
\end{bmatrix}
\]

Min and Max Sigma Values:

\[
\sigma_{\text{max}}^2 = \frac{\left(\sigma_v^2 + \sigma_n^2\right) + \sqrt{\left(\sigma_v^2 + \sigma_n^2\right)^2 - 4\left(\sigma_v^2\sigma_n^2 - \sigma_{vn}^2\right)}}{2}
\]

\[
\sigma_{\text{min}}^2 = \frac{\left(\sigma_v^2 + \sigma_n^2\right) - \sqrt{\left(\sigma_v^2 + \sigma_n^2\right)^2 - 4\left(\sigma_v^2\sigma_n^2 - \sigma_{vn}^2\right)}}{2}
\]

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

With the vertical (or “up”) variance known \( (\sigma_u) \), 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>47</td>
<td>Generic Flag Data 01</td>
<td>None</td>
<td>uint8</td>
<td>uint8</td>
</tr>
</tbody>
</table>

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

Example Value: 49
Example LS Packet: [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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>Security Local Metadata Set</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td>Use ST0102 US key for Local Sets.</td>
<td>None</td>
<td>Set</td>
<td>Set</td>
</tr>
<tr>
<td>Notes</td>
<td>Conversion Formula</td>
<td>x</td>
<td>x</td>
<td>-</td>
</tr>
</tbody>
</table>

- 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.

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>× K][L][V] = [0d48][0dx][x]</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>US Key</th>
<th>Security Local Metadata Set</th>
<th>ESD Digraph</th>
</tr>
</thead>
<tbody>
<tr>
<td>US Name</td>
<td>Use ST0102 US key for Local Sets.</td>
<td>ESD Name</td>
</tr>
<tr>
<td>Notes</td>
<td>Notes</td>
<td></td>
</tr>
</tbody>
</table>

- US Conversion
  - To US: ×
  - To LS: ×

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>49</td>
<td>Differential Pressure</td>
<td>Millibar</td>
<td>0..5000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

### Notes
- 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

### Conversion Formula

\[
\text{LS\_dec} = \left( \frac{\text{LS\_range\_uint}}{65535} \right) \times \text{LS\_uint} \\
\text{LS\_49\_dec} = \left( \frac{5000}{65535} \right) \times \text{LS\_49}
\]

### Example Value

1191.958 mbar

### Example LS Packet

\[ [K][L][V] = [0d49][0d2][0x3D 07] \]

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 (\( p_d = \text{impact pressure} - \text{static pressure} = p_i - p_s \)), and \( \rho \) is the density of the fluid outside the item.
8.50 Tag 50: Platform Angle of Attack Conversion

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>Platform Angle of Attack</td>
<td>Degrees</td>
<td>+/- 20</td>
<td>int16</td>
</tr>
</tbody>
</table>

**Notes**
- 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.

**Conversion Formula**

$$\text{LS}_{\text{dec}} = \left( \frac{\text{LS}_{\text{range}} \cdot \text{LS}_{\text{int}}}{2^{15}} \right)$$

$$\text{LS}_{50\_\text{dec}} = \left( \frac{40}{65535} \cdot \text{LS}_{50} \right)$$

**Example Value**

-8.670177 Degrees

**Example LS Packet**

[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.

![Platform Angle of Attack](image)

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>51</td>
<td>Platform Vertical Speed</td>
<td>Meters/Second</td>
<td>+/- 180</td>
<td>int16</td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula

\[
\text{LS dec} = \left( \frac{\text{LS range}}{\text{int range}} \right) \times \text{LS int}
\]

Example Value

-61.88693 m/s

Example LS Packet

[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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>52</td>
<td>Platform Sideslip Angle</td>
<td>Degrees</td>
<td>+/- 20</td>
<td>int16</td>
</tr>
</tbody>
</table>

**Notes**

- The sideslip angle is the angle between the platform longitudinal axis and relative wind.
- Positive angles to right wing, neg to left.
- Map $-(2^{15}-1)\ldots(2^{15}-1)$ to +/-20.
- Use $-(2^{15})$ as an "out of range" indicator.
- $-(2^{15}) = 0x8000$.
- Res: ~610 micro deg.

**Conversion Formula**

\[
\text{LS}_\text{dec} = \left( \frac{\text{LS}_\text{range}}{\text{int}_\text{range}} \times \text{LS}_\text{int} \right)
\]

\[
\text{LS}_{52}_{\text{dec}} = \left( \frac{40}{65534} \times \text{LS}_{52} \right)
\]

**Example Value**

-5.062475 Degrees

**Example LS Packet**

[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](image)

Note that the int16 used in the LS value is encoded using two’s complement.
### 8.53 Tag 53: Airfield Barometric Pressure Conversion

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>Airfield Barometric Pressure</td>
<td>Millibar</td>
<td>0..5000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

#### Notes
- 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

#### Conversion Formula

<table>
<thead>
<tr>
<th>LS_dec = (LS_range * LS_uint)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS_53_dec = (5000 * LS_53)</td>
</tr>
</tbody>
</table>

#### Example Value

2088.96 mbar

#### Example LS Packet

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

#### Notes
- 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.

#### Conversion Formula
- \( \text{LS}_\text{dec} = \left( \frac{\text{uint}_\text{range} \times \text{LS}_\text{int}}{\text{LS}_\text{int}} \right) - \text{Offset} \)
- \( \text{LS}_54\_\text{dec} = \left( \frac{19900}{\text{LS}_54} \right) - 900 \)

#### Example Value
8306.806 Meters

#### Example LS Packet
\([K][L][V] = [0d54][0d2][0x76 70]\)

### 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>55</td>
<td>Relative Humidity</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 01 09 00 00 00</td>
<td>Percent</td>
<td>0..100</td>
<td>uint8</td>
</tr>
</tbody>
</table>

Notes
- Relative Humidity at aircraft location.
- Map 0..(2^8-1) to 0..100.
- Resolution: ~0.4%.

Conversion Formula
\[
\text{LS}_{\text{dec}} = \left( \frac{\text{LS\_range}}{\text{uint\_range}} \right) \times \text{LS\_uint} \\
\text{LS\_55\_dec} = \left( \frac{100}{255} \right) \times \text{LS\_55}
\]

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

![Table](https://via.placeholder.com/150)

Notes
- 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.

Conversion Formula
-\[ \text{LS\_dec} = \text{LS\_int} \]
-\[ \text{LS\_56\_dec} = \text{round}(\text{LS\_56}) \]

Example Value
140 m/s

Example LS Packet
\[ \text{[K][L][V]} = [0d56][0d1][0x8C] \]

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 |
| LS Name | Ground Range |
| US Mapped Key | 06 0E 2B 34 01 01 01 06 01 01 06 00 00 00 |

**Units** | **Range** | **Format**
--- | --- | ---
Meters | 0..5,000,000 | uint32

**Notes**
- 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 millimeters.

**Conversion Formula**
\[
\text{LS}_\text{dec} = \left( \text{LS}_{\text{range}} \times \text{LS}_{\text{uint}} \right)
\]
\[
\text{LS}_{57}\_\text{dec} = \left( \frac{5000000}{4294967295} \times \text{LS}_57 \right)
\]

**Example Value**
3506979 Meters

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

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

**ESD Conversion**
- **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

| LS Tag | 58 |
| LS Name | Platform Fuel Remaining |
| US Mapped Key | 06 0E 2B 34 01 01 01 01 0E 01 01 07 00 00 00 |
| Units | Kilogram |
| Range | 0.10,000 |
| Format | uint16 |

**Notes**
- 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.

**Conversion Formula**

\[
\begin{align*}
\text{LS}_{\text{dec}} &= \left(\frac{\text{LS}_{\text{range}}}{\text{uint}_\text{range}} \times \text{LS}_{\text{uint}}\right) \\
\text{LS}_{58}_{\text{dec}} &= \left(\frac{10000}{65535} \times \text{LS}_{58}\right)
\end{align*}
\]

**Example Value**

6420.539 kg

**Example LS Packet**

[K][L][V] = [0d58][0d32][0xA4 5D]

**US Key**

x

**US Name**

x

**ESD Digraph**

Fr

**ESD Name**

Platform Fuel Remaining

**Units** | Range | Format
---|---|---
Pounds | 0..99,999 | N

**Notes**
- x

- Remaining fuel on airborne platform. Metered as fuel weight remaining.

**US Conversion**

- **To US:**
  - x

- **To LS:**
  - x

**ESD Conversion**

\[
\begin{align*}
\text{ESD}_{\text{dec}} &= \left(\frac{10000}{65535} \times \text{LS}_{\text{uint}}\right) \times 2.046226218lbs \over 1kg
\end{align*}
\]

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>59</td>
<td>Platform Call Sign</td>
<td>06 0E 2B 34 01 01 01 0E 01 04 01 01 00 00 00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notes</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Call Sign of platform or operating unit.</td>
<td>x</td>
</tr>
<tr>
<td>- Value field is Free Text.</td>
<td>x</td>
</tr>
</tbody>
</table>

Example Value

<table>
<thead>
<tr>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOP GUN [K][L][V] = [0d59][0d7][0x54 4F 50 20 47 55 4E]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>US Key</th>
<th>US Name</th>
<th>ESD Digraph</th>
<th>ESD Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>x</td>
<td>Cs</td>
<td>Platform Call Sign</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notes</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>- x</td>
<td>- First nine characters of the Call Sign of a group or squadron.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>US Conversion</th>
<th>ESD Conversion</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

To US:
- 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

| LS Tag | 60 |
| 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: | x |
| - [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 | |

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

| US Key | x |
| US Name | x |

| ESD Digraph | Wl |
| ESD Name | Weapon Load |

| Units | Range | Format | Units | Range | Format |
| Notes | x | x | Notes | x | x |

| US Conversion | ESD Conversion |
| To US: | x |
| To LS: | x |
| To ESD: | 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

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

Notes
- 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

Conversion Formula

Example Value

| US Key | x |
| US Name | x |
| ESD Digraph | WF |
| ESD Name | Weapon Fired |

Example LS Packet

186 [K][L][V] = {0d610d20xBA}

Notes
- x

To US:
- x

To LS:
- 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>62</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td>Laser PRF Code</td>
</tr>
<tr>
<td>US Mapped Key</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 02 02 01 00 00 00</td>
</tr>
<tr>
<td>Units</td>
<td>None</td>
</tr>
<tr>
<td>Range</td>
<td>0..65535</td>
</tr>
<tr>
<td>Format</td>
<td>uint16</td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula

\[ \text{Example Value} \]

Example LS Packet

50895

\[ \{K\|L\|V\} = \{0d62\|0d2\|0xC6 CF\} \]

Conversion

To US:
- Convert LS uint to ASCII.

To ESD:
- Convert ASCII to LS uint.

US Conversion

ESD Conversion

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>63</td>
<td>Sensor Field of View Name</td>
<td>06 0E 2B 34 01 01 01 0E 01 02 02 02 00 00 00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Notes
- 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

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>209</td>
<td>[K][L][V] = [0d63][0d1][0xD1]</td>
</tr>
</tbody>
</table>

### 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

### Example Value
- US Key: x
- US Name: x

### ESD Digraph
- ESD Name: Sensor Field of View Name

### Notes
- x

### US Conversion
- To US: Convert LS uint to ASCII.
- To LS: Convert ASCII to LS uint.
## 8.64 Tag 64: Platform Magnetic Heading Conversion

### LS Tag
- **Tag**: 64
- **Name**: Platform Magnetic Heading
- **Key**: 06 0E 2B 34 01 01 01 01 0E 01 01 08 00 00 00

**Units** | **Range** | **Format** |
--- | --- | --- |
Degrees | 0..360 | uint16 |

### Notes
- 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.

### Conversion Formula
- **LS:**
  
  \[ \text{LS}\_\text{dec} = \left( \frac{\text{LS}\_\text{range}}{65535} \right) \times \text{LS}\_\text{uint} \]

- **LS 64:**
  
  \[ \text{LS}\_64\_\text{dec} = \left( \frac{360}{65535} \right) \times \text{LS}\_64 \]

### Example Value
- **311.8682 Degrees**

### Example LS Packet
- 
  \[(K)\{L}\{V\} = \{0d64\}{0d2}\{0xDD C5\}

#### US Key
- **x**

#### US Name
- **Mh**

#### ESD Digraph
- **x**

#### ESD Name
- **Platform Magnetic Heading**

### Notes
- Aircraft magnetic heading angle. Relative between fuselage chord line and Magnetic North.

### US Conversion
- **x**

### ESD Conversion
- **x**

### To US:
- **x**

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

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

### Example Magnetic Heading

![Figure 8-20: Magnetic Heading](image)

---

23 October 2014  Motion Imagery Standards Board  114
8.65 Tag 65: UAS LS Version Number Conversion

| LS Tag | 65 |
| LS Name | UAS LS Version Number |
| US Mapped Key | 06 0E 2B 34 01 01 01 01 0E 01 02 03 03 00 00 00 |
| Units | Range | Format |
| Number | 0..255 | uint8 |

Notes
- 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.

| Example Value | Example LS Packet |
| Version 232 | [K][L][V] = [0d65][0d1][0xE8] |

| US Key | x |
| US Name | x |

Notes
- x

| Units | Range | Format | ESD Digraph | ESD Name | ESD ICD Version |
| Number | 0..99 | NN |

Notes
- x
- Version of the ESD System used to encode ESD Data.
- 0 corresponds to documents ASI-119 and ASI-209.
- 1..99 corresponds to document revisions ST0601.1 thru ST0601.99.

| US Conversion | ESD Conversion |
| x |

Notes
To US:
- x

Notes
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

| LS Tag | 66 |
| 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 |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD</td>
<td>TBD</td>
<td>TBD</td>
</tr>
</tbody>
</table>

Notes
- Covariance Matrix of the error associated with a targeted location.
- Details TBD.

Conversion Formula
TBD
TBD

Example Value
\[ x^{[K][L][V]} = [0d66][0dTBD][x] \]

8.66.1 Example Target Location Covariance Matrix
Details TBD
8.67 Tag 67: Alternate Platform Latitude Conversion

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

Notes
- Alternate Platform Latitude. Represents latitude of platform connected with UAS.
- Based on WGS84 ellipsoid.
- Map -(2^31-1) to +/90.
- Use -(2^31) as an "error" indicator.
- Resolution: ~42 nano degrees.

Conversion Formula
\[
\text{LS\_dec} = \left( \frac{\text{LS\_range}}{\text{int\_range}} \right) \\
\text{LS\_67\_dec} = \left( \frac{180}{2^{31}-1} \right) \times \text{LS\_67}
\]

Example Value

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>-86.041207348947</td>
<td>[K][L][V] = [0d67][0d4][0x85 A1 5A 39]</td>
</tr>
</tbody>
</table>

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 directdatalink 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>68</td>
<td>Alternate Platform Longitude</td>
<td>06 0E 2B 34 01 01 01</td>
<td>Degrees</td>
<td>+/- 180</td>
<td>int32</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0E 01 01 01 15 00 00 00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula
- LS_dec = (LS_range / int_range) * LS_int
- LS_68_dec = (360 / 2^31-1) * LS_68

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>69</td>
<td>Alternate Platform Altitude</td>
<td>Meters</td>
<td>-900..19000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>US Mapped Key</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34</td>
<td>01 01 01 01</td>
<td>0E 01 01 01</td>
<td>16 00 00 00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Notes**

- Altitude of alternate platform as measured from Mean Sea Level (MSL).
- Represents altitude of platform connected with UAS.
- Map 0..(2^16-1) to -900..19000 meters.
- 1 meter = 3.2808399 feet.
- Resolution: ~0.3 meters.

**Conversion Formula**

\[
\text{LS}_{\text{dec}} = \left( \frac{\text{LS}_{\text{range}}}{\text{uint}_{\text{range}}} \right) \times \text{LS}_{\text{uint}} - \text{Offset}
\]

\[
\text{LS}_{69\text{dec}} = \left( \frac{19900}{65535} \right) \times \text{LS}_{69} - 900
\]

**Example Value**

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.445334 Meters</td>
<td>[K][L][V] = [0d69][0d2][0x0B B3]</td>
</tr>
</tbody>
</table>

### 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>Alternate Platform Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>70</td>
<td></td>
<td>String</td>
<td>1..127</td>
<td>ISO 646</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LS Name</th>
<th>US Mapped Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 01 01 01 01</td>
<td></td>
</tr>
<tr>
<td>0E 01 01 01 17 00 00 00</td>
<td></td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula
- \( [K][L][V] = [0d70][0d6][0x41 50 41 43 48 45] \)

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>71</td>
<td>Alternate Platform Heading</td>
<td>Degrees</td>
<td>0..360</td>
<td>uint16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>71</td>
<td></td>
<td>06 0E 2B 34</td>
<td>- Heading angle of alternate platform connected to UAS. Relative</td>
</tr>
<tr>
<td></td>
<td></td>
<td>01 01 01 01</td>
<td>between longitudinal axis and True North measured in the horizontal</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0E 01 01 01</td>
<td>plane.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>18 00 00 00</td>
<td>- Map 0..(2^16-1) to 0..360.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Resolution: ~5.5 milli degrees.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS_dec = \left(\frac{\text{LS_range}}{\text{int_range}} \times \text{LS_int}\right)</td>
</tr>
<tr>
<td>LS_71_dec = \left(\frac{360}{65535} \times \text{LS}_71\right)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>32.60242 Degrees</td>
<td>[K][L][V] = [0d71][0d2][0x17 2F]</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>72</td>
<td>Event Start Time - UTC</td>
<td>Use EG0104 US Key</td>
<td>Microseconds</td>
<td>0..(2^64-1)</td>
<td>uint64</td>
</tr>
</tbody>
</table>

Notes
- 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.

Example Value
April 16, 1995. 13:44:54

Example LS Packet
[K][L][V] = [0d72][0d8][0x00 02 D5 CF 4D DC 9A 35]

US Key
06 0E 2B 34 01 01 01
07 02 01 02 07 01 00 00

US Name
Event Start Date Time - UTC

Notes
- The absolute beginning date and time of the project, mission, scene, editing event, license, publication etc.
- Formatted text as: 'YYYYMMDDhhmmss'

To US:
- Convert uint64 to formatted string.

To ESD:
- x

To LS:
- Convert formatted string to uint64.

ESD Conversion
- 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.

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>73</td>
<td>RVT Local Set</td>
<td>None</td>
<td>Set</td>
<td>Set</td>
</tr>
</tbody>
</table>

Use ST0806 RVT LS 16-byte Key.

Notes:
- 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.

Conversion Formula

Example Value

<table>
<thead>
<tr>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>[K][L][V] = [0d73][0d</td>
</tr>
</tbody>
</table>

Example LS Packet

<table>
<thead>
<tr>
<th>US Key</th>
<th>US Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 02 0B 01 01</td>
<td></td>
</tr>
<tr>
<td>0E 01 03 01 02 00 00 00</td>
<td></td>
</tr>
<tr>
<td>Remote Video Terminal Local Set</td>
<td></td>
</tr>
</tbody>
</table>

Notes:
- x

ESD Digraph

ESD Name

Conversion:

<table>
<thead>
<tr>
<th>US Conversion</th>
<th>ESD Conversion</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

To US:
- 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 |
| LS Name | VMTI Data 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. |

| Example Value | Example LS Packet |
| 06 0E 2B 34 | [K][L][V] = [0d74][0d|x] |

| US Key | 06 0E 2B 34 02 0B 01 01 0E 01 03 03 06 00 00 00 |
| US Name | Video Moving Target Indicator Local Set |

| US Conversion | ESD Conversion |
| - x | - 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>75</td>
<td>Sensor Ellipsoid Height</td>
<td>Meters</td>
<td>-900..19000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

**Notes**
- 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.

**Conversion Formula**

\[
\text{LS}_{\text{dec}} = \left( \frac{\text{LS}_{\text{range}} \times \text{LS}_{\text{uint}}}{\text{uint}_{\text{range}}} \right) - \text{Offset}
\]

\[
\text{LS}_{75\text{dec}} = \left( \frac{19900 \times \text{LS}_{75}}{65535} \right) - 900
\]

**Example Value**

14190.72 Meters

**Example LS Packet**

[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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>76</td>
<td>Alternate Platform Ellipsoid Height</td>
<td>Meters</td>
<td>-900..19000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>US Mapped Key</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 01 01 01 01</td>
<td>LS_dec = ( \frac{\text{LS_range}}{\text{uint_range}} \times \text{LS_uint} ) - Offset</td>
</tr>
<tr>
<td>0E 01 02 01 82 48 00 00</td>
<td>LS_76_dec = ( \frac{19900}{65535} \times \text{LS_76} ) - 900</td>
</tr>
</tbody>
</table>

#### Notes
- 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.

#### Example Value

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.445334 Meters</td>
<td>[K][L][V] = [0d76][0d2][0x0B B3]</td>
</tr>
</tbody>
</table>

### 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>Operational Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td>77</td>
</tr>
<tr>
<td>US Mapped Key</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 01 03 21 00 00 00</td>
</tr>
<tr>
<td>Units</td>
<td>Range</td>
</tr>
<tr>
<td>None</td>
<td>None</td>
</tr>
</tbody>
</table>

Notes
- Indicates the mode of operations of the event portrayed in metadata. Enumerated.
- 0x00 = "Other"
- 0x01 = "Operational"
- 0x02 = "Training"
- 0x03 = "Exercise"
- 0x04 = "Maintenance"
- 0x05 = "Test"

Conversion Formula

Example Value: x
Example LS Packet: [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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>78</td>
<td>Frame Center Height Above Ellipsoid</td>
<td>Meters</td>
<td>-900..19000</td>
<td>uint16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notes</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Frame Center Ellipsoid Height as measured from the reference WGS84 Ellipsoid.</td>
<td>LS_dec = \left(\frac{\text{LS_range}}{\text{uint_range}} \times \text{LS_uint}\right) - \text{Offset}</td>
</tr>
<tr>
<td>- Map 0..(2^16-1) to -900..19000 meters.</td>
<td>LS_78_dec = \left(\frac{19900}{65535} \times \text{LS_78}\right) - 900</td>
</tr>
<tr>
<td>- 1 meter = 3.2808399 feet.</td>
<td></td>
</tr>
<tr>
<td>- Resolution: ~0.3 meters.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.445334 Meters</td>
<td>[K][L][V] = [0d78][0d2][0x0B B3]</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>79</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td>Sensor North Velocity</td>
</tr>
<tr>
<td>US Mapped Key</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 02 02 7E 00 00 00</td>
</tr>
<tr>
<td>Units</td>
<td>Meters/Sec</td>
</tr>
<tr>
<td>Range</td>
<td>+/-327</td>
</tr>
<tr>
<td>Format</td>
<td>int16</td>
</tr>
</tbody>
</table>

Notes
- 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.

Conversion Formula
\[
\text{LS}_{\text{dec}} = \left( \frac{\text{LS}_{\text{range}} \times \text{LS}_{\text{int}}}{(2^{15})} \right) \\
\text{LS}_{79} = \left( \frac{654 \times \text{LS}_{79}}{65534} \right)
\]

Example Value

Example LS Packet

<table>
<thead>
<tr>
<th>x</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>
### 8.80 Tag 80: Sensor East Velocity Conversion

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>Sensor East Velocity</td>
<td>06 0E 2B 34 01 01 01 01 0E 01 02 02 7F 00 00 00</td>
<td>Meters/Sec</td>
<td>+/-327</td>
<td>int16</td>
</tr>
</tbody>
</table>

#### Notes
- 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.

#### Conversion Formula

| LS_dec = (LS_range * LS_int) | LS_80 = (654 * LS_80) |

#### Example Value

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>[K][L][V] = [0d80][0dx][x]</td>
</tr>
</tbody>
</table>

### 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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>81</td>
<td>Image Horizon Pixel Pack</td>
<td>Pack</td>
<td>Pack</td>
<td>Pack</td>
</tr>
</tbody>
</table>

| Key     | 06 0E 2B 34 02 05 01 01 | 0E 01 03 02 08 00 00 00 |

### Notes
- `<tag 81><length>`
- `<start x0, start y0 // point p0`
- `end x1, end y1 // point p1`
- `start lat, start lon`
- `end lat, end lon`
- `>`

### Conversion Formula
- **Conversion Formula**

\[
\text{Conversion Formula} \to \text{See Notes below.}
\]

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>{K}[L][V] = [0d81][0dx][x]</td>
</tr>
</tbody>
</table>

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 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.

![Pixel Frame Examples](image)

**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:

![Image Horizon Pixel Pack Example](image)

**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

<table>
<thead>
<tr>
<th>Local Set Key</th>
<th>Name</th>
<th>Notes</th>
<th>Units/Range</th>
<th>Format</th>
<th>Len</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 - 02 05 01 01 - 0E 01 03 02 - 08 00 00 00</td>
<td>Image Horizon Pixel Pack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(CRC 37658)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Constituent Elements</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key</td>
<td>Name</td>
<td>Notes</td>
<td>Units/Range</td>
<td>Format</td>
<td>Len</td>
</tr>
<tr>
<td>06 0E 2B 34</td>
<td>Start x0</td>
<td>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.</td>
<td>Percent [0..100]</td>
<td>Uint8</td>
<td>1</td>
</tr>
<tr>
<td>01 01 01 01</td>
<td>0E 01 01 02</td>
<td>09 01 00 00</td>
<td>(CRC 3334)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>06 0E 2B 34</td>
<td>Start y0</td>
<td>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.</td>
<td>Percent [0..100]</td>
<td>Uint8</td>
<td>1</td>
</tr>
<tr>
<td>01 01 01 01</td>
<td>0E 01 01 02</td>
<td>09 02 00 00</td>
<td>(CRC 21590)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>06 0E 2B 34</td>
<td>End x1</td>
<td>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.</td>
<td>Percent [0..100]</td>
<td>Uint8</td>
<td>1</td>
</tr>
<tr>
<td>01 01 01 01</td>
<td>0E 01 01 02</td>
<td>09 03 00 00</td>
<td>(CRC 25446)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>06 0E 2B 34</td>
<td>End y1</td>
<td>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.</td>
<td>Percent [0..100]</td>
<td>Uint8</td>
<td>1</td>
</tr>
<tr>
<td>01 01 01 01</td>
<td>0E 01 01 02</td>
<td>09 04 00 00</td>
<td>(CRC 59126)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>06 0E 2B 34</td>
<td>Start Latitude</td>
<td>The Latitude of the Start point (x0,y0) on the image border. Based on WGS84 ellipsoid. Map -2^31,-2^31 to +/-90. Use (-2^31) as an &quot;error&quot; indicator. Optional (but recommended).</td>
<td>Degrees [-90..+90]</td>
<td>Int32</td>
<td>4</td>
</tr>
<tr>
<td>01 01 01 01</td>
<td>0E 01 01 02</td>
<td>09 05 00 00</td>
<td>(CRC 53702)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>06 0E 2B 34</td>
<td>Start Longitude</td>
<td>The Longitude of the Start point (x0,y0) on the image boarder. Based on WGS84 ellipsoid. Map -2^31,-2^31 to +/-180. Use (-2^31) as an &quot;error&quot; indicator. Optional (but recommended).</td>
<td>Degrees [-180..+180]</td>
<td>Int32</td>
<td>4</td>
</tr>
<tr>
<td>01 01 01 01</td>
<td>0E 01 01 02</td>
<td>09 06 00 00</td>
<td>(CRC 34966)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>06 0E 2B 34</td>
<td>End Latitude</td>
<td>The Latitude of the End point (x1,y1) on the image boarder. Based on WGS84 ellipsoid. Map -2^31,-2^31 to +/-90. Use (-2^31) as an &quot;error&quot; indicator. Optional (but recommended).</td>
<td>Degrees [-90..+90]</td>
<td>Int32</td>
<td>4</td>
</tr>
<tr>
<td>01 01 01 01</td>
<td>0E 01 01 02</td>
<td>09 07 00 00</td>
<td>(CRC 49062)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>06 0E 2B 34</td>
<td>End Longitude</td>
<td>The Longitude of the End point (x1,y1) on the image boarder. Based on WGS84 ellipsoid. Map -2^31,-2^31 to +/-180. Use (-2^31) as an &quot;error&quot; indicator. Optional (but recommended).</td>
<td>Degrees [-180..+180]</td>
<td>Int32</td>
<td>4</td>
</tr>
<tr>
<td>01 01 01 01</td>
<td>0E 01 01 02</td>
<td>09 08 00 00</td>
<td>(CRC 37783)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### 8.82 Tag 82: Corner Latitude Point 1 (Full) Conversion

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>82</td>
<td>Corner Latitude Point 1 (Full)</td>
<td>Use EG0104 US Key</td>
<td>Degrees</td>
<td>+/- 90</td>
<td>int32</td>
</tr>
</tbody>
</table>

**Notes**
- 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.

**Conversion Formula**

\[
\text{LS}_{\text{dec}} = \left( \frac{\text{LS}_{\text{range}} \times \text{LS}_{\text{int}}}{180} \right)
\]

\[
\text{LS}_{82\text{ dec}} = \left( \frac{180}{4294967294} \times \text{LS}_82 \right)
\]

**Example Value**

-10.579637999887 Corrected Degrees

<table>
<thead>
<tr>
<th>US Key</th>
<th>US Name</th>
<th>ESD Digraph</th>
<th>ESD Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34</td>
<td>Corner Latitude Point 1 (Decimal Degrees)</td>
<td>SAR Latitude 4</td>
<td></td>
</tr>
<tr>
<td>01 01 01 03</td>
<td>07 01 02 01</td>
<td>03 07 01 00</td>
<td></td>
</tr>
</tbody>
</table>

**US Conversion**

\[
\text{US}_{\text{dec}} = \left( \frac{180}{4294967294} \times \text{LS}_{\text{int}} \right)
\]

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

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

**ESD Conversion**

\[
\text{ESD}_{\text{dec}} = \left( \frac{180}{4294967294} \times \text{LS}_{82} \right)
\]

**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).
Value is encoded using two’s complement.

Figure 8-24: Offset Corner Point 1
8.83 Tag 83: Corner Longitude Point 1 (Full) Conversion

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>83</td>
<td>Corner Longitude Point 1 (Full)</td>
<td>Degrees</td>
<td>+/- 180</td>
<td>int32</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notes on Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Frame Longitude for upper left corner.</td>
</tr>
<tr>
<td>- Full Range.</td>
</tr>
<tr>
<td>- Based on WGS84 ellipsoid.</td>
</tr>
<tr>
<td>- Map -(-2^31-1)...-(-2^31-1) to +/-180.</td>
</tr>
<tr>
<td>- Use -(-2^31) as an &quot;error&quot; indicator.</td>
</tr>
<tr>
<td>- -(-2^31) = 0x80000000.</td>
</tr>
<tr>
<td>- Resolution: ~84 nano degrees.</td>
</tr>
</tbody>
</table>

**Example Value**

29.127367798633 Corrected Degrees

**Example LS Packet**

(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

**US Name**

Corner Longitude Point 1 (Decimal Degrees)

**ESD Digraph**

Rh

**ESD Name**

SAR Longitude 4

<table>
<thead>
<tr>
<th>Notes</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Longitude coordinate of corner 1 of an image or bounding rectangle.</td>
<td></td>
</tr>
<tr>
<td>- Positive (+) is eastern hemisphere.</td>
<td></td>
</tr>
<tr>
<td>- Negative (-) is western hemisphere.</td>
<td></td>
</tr>
</tbody>
</table>

**US Conversion**

US_dec = (360/4294967294 * LS_int)

**ESD Conversion**

ESD_dec = (360/4294967294 * LS_int)

To US:

- US = (double)(360/0xFFFFFFFFF * LS)

To LS:

- LS = (int32)round(0xFFFFFFFFF/360 * US)

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>84</td>
<td>Corner Latitude Point 2 (Full)</td>
<td>Degrees</td>
<td>+/- 90</td>
<td>int32</td>
</tr>
<tr>
<td></td>
<td>Use EG0104 US Key</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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
- \[
    LS_{\text{dec}} = \left( \frac{\text{LS range}}{\text{int range}} \right) \times \text{LS int}
    \]
- \[
    LS_{84 \text{ dec}} = \left( \frac{180}{4294967294} \right) \times \text{LS 84}
    \]

Example Value
-10.5661816260963 Corrected

Example LS Packet
[K][L][V] = [0d84][0d2][0xD7 65]

US Conversion
- \[
    \text{US}_{\text{dec}} = \left( \frac{180}{4294967294} \right) \times \text{LS int}
    \]
- \[
    \text{To US:} \quad \text{US} = \text{(double)}(180/0xFFFFFFFFE \times \text{LS})
    \]
- \[
    \text{To LS:} \quad \text{LS} = \text{(int32)}\text{round}(0xFFFFFFFFE/180 \times \text{US})
    \]

ESD Conversion
- \[
    \text{ESD}_{\text{dec}} = \left( \frac{180}{4294967294} \right) \times \text{LS int}
    \]
- \[
    \text{To ESD:} \quad \text{Convert LS to decimal.}
    \]
- \[
    \text{Convert decimal to ASCII.}
    \]
- \[
    \text{To LS:} \quad \text{Convert ASCII to decimal.}
    \]
- \[
    \text{Map decimal to int32.}
    \]

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).
Value is encoded using two’s complement.
8.85 Tag 85: Corner Longitude Point 2 (Full) Conversion

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>85</td>
<td>Corner Longitude Point 2 (Full)</td>
<td>Degrees</td>
<td>+/- 180</td>
<td>int32</td>
</tr>
</tbody>
</table>

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

**Conversion Formula**
- LS_dec = \( \frac{\text{LS range}}{\text{int range}} \times \text{LS int} \)
- LS_85_dec = \( \frac{360}{2^{31} - 1} \times \text{LS_85} \)

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>29.140824172424 Corrected</td>
<td>{K}[L][V] = {0d85}{0d2}{0xE2 E0}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>US Key</th>
<th>ESD Digraph</th>
<th>ESD Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 0C 01 00</td>
<td>SAR Longitude 1</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LS dec</th>
<th>US Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 0C 01 00</td>
<td>Corner Longitude Point 2 (Decimal Degrees)</td>
</tr>
</tbody>
</table>

**US Conversion**
- US_dec = \( \frac{360}{2^{31} - 1} \times \text{LS int} \)

**ESD Conversion**
- ESD_dec = \( \frac{360}{2^{31} - 1} \times \text{LS int} \)

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

**Notes**
- The longitude of the upper right corner of the SAR image box.

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

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

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>86</td>
<td>Corner Latitude Point 3 (Full)</td>
<td>Degrees</td>
<td>+/- 90</td>
<td>int32</td>
</tr>
</tbody>
</table>

Use EGO104 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.

Example Value:
-10.5527275411938 Corrected

degrees

Example LS Packet:
[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
Corner Latitude Point 3
(Decimal Degrees)

US Name:
Corner Latitude Point 3

ESD Digraph:
Rc

ESD Name:
SAR Latitude 2

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

Conversion Formula:

\[ LS\_dec = \left( \frac{LS\_range}{1\text{nt}\_range} \right) \cdot LS\_int \]

\[ LS\_86\_dec = \left( \frac{180}{4294967294} \right) \cdot LS\_86 \]

US Conversion:

\[ US\_dec = \left( \frac{180}{4294967294} \right) \cdot LS\_int \]

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

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

ESD Conversion:

\[ ESD\_dec = \left( \frac{180}{4294967294} \right) \cdot LS\_int \]

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>LS Mapped Key</th>
<th>Notes</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>87</td>
<td>Corner Longitude Point 3 (Full)</td>
<td>Use EG0104 US Key</td>
<td>- Frame Longitude for lower right corner.</td>
<td>( LS_{\text{dec}} = \frac{LS_{\text{range}}}{\text{int32}} \times LS_{\text{int}} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Full Range.</td>
<td>( LS_{87\text{dec}} = \frac{360}{4294967294} \times LS_{87} )</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Based on WGS84 ellipsoid.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Map ((-2^\circ31\text{-}1)\ldots(2^\circ31\text{-}1) to +/-180.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Use (-2^\circ31) as an &quot;error&quot; indicator.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- (-2^\circ31) = 0x80000000.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Resolution: ~84 nano degrees.</td>
<td></td>
</tr>
</tbody>
</table>

### Example Value

<table>
<thead>
<tr>
<th>29.1542762573265 Corrected Degrees</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>([K][L][V] = [0d87][0d2][0xF9 D6] )</td>
</tr>
</tbody>
</table>

### Example LS Packet

<table>
<thead>
<tr>
<th>US Key</th>
<th>US Name</th>
<th>ESD Digraph</th>
<th>ESD Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>06 0E 2B 34 01 01 01 03 07 01 02 01 03 0D 01 00</td>
<td>Corner Longitude Point 3 (Decimal Degrees)</td>
<td>SAR Longitude 2</td>
<td></td>
</tr>
</tbody>
</table>

### 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

\[
\text{US}_{\text{dec}} = \frac{360}{4294967294} \times \text{LS}_{\text{int}} \\
\text{To US:} - \text{US} = (\text{double})(360/0xFFFFFFFFF \times \text{LS})
\]

### ESD Conversion

\[
\text{ESD}_{\text{dec}} = \frac{360}{4294967294} \times \text{LS}_{\text{int}} \\
\text{To ESD:} - \text{Convert LS to decimal.} \\
\text{Convert decimal to ASCII.}
\]

### To LS

- \( \text{LS} = \) \text{(int32)} round(0xFFFFFFFFF/360 \times \text{US})
- \text{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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>88</td>
<td>Corner Latitude Point 4 (Full)</td>
<td>Degrees</td>
<td>+/- 90</td>
<td>int32</td>
</tr>
</tbody>
</table>

**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**

\[
\text{US} = (\text{double})(180/0xFFFFFFFFFE \times \text{LS})
\]

\[
\text{LS}_{88\_\text{dec}} = \left( \frac{180}{4294967294} \right) \times \text{LS}_{88}
\]

**Example Value**
-10.5392711674031 Corrected Degrees

**Example LS Packet**
\[
[K][L][V] = [\text{0d88}][\text{0d2}][\text{0x05 52}]
\]

**US Key**
- 06 0E 2B 34  01 01 01 03
- 07 01 02 01  03 0A 01 00
- Corner Latitude Point 4 (Decimal Degrees)

**US Name**
- Corner Latitude Point 4

**ESD Digraph**
- SAR Latitude 3

**ESD Name**
- SAR Latitude 3

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

**US Conversion**

\[
\text{US\_dec} = \left( \frac{180}{4294967294} \right) \times \text{LS\_int}
\]

**To US:**
- \( \text{US} = (\text{double})(180/0xFFFFFFFFFE \times \text{LS}) \)

**To LS:**
- \( \text{LS} = (\text{int32})\text{round}(0xFFFFFFFFFE/180 \times \text{US}) \)

**ESD Conversion**

\[
\text{ESD\_dec} = \left( \frac{180}{4294967294} \right) \times \text{LS\_int}
\]

**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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>89</td>
<td>Corner Longitude Point 4 (Full)</td>
<td>Degrees</td>
<td>+/- 180</td>
<td>int32</td>
</tr>
</tbody>
</table>

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

Conversion Formula

\[ LS_{dec} = \frac{LS_{range}}{0x7FFFFFFF} \times LS_{int} \]

\[ LS_{89}_{dec} = \frac{360}{4294967294} \times LS_{89} \]

Example Value

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>29.167346311172 Corrected</td>
<td>[K][L][V] = [0d89][0d2][0x10 CD]</td>
</tr>
</tbody>
</table>

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

US Conversion

\[ US_{dec} = \frac{360}{4294967294} \times LS_{int} \]

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

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

ESD Conversion

\[ ESD_{dec} = \frac{360}{4294967294} \times LS_{int} \]

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 |
| LS Name | Platform Pitch Angle (Full) |
| US Mapped Key | Use EG0104 US Key |

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>+/- 90</td>
<td>int32</td>
</tr>
</tbody>
</table>

Notes:
- Aircraft pitch angle. Angle between longitudinal axis and horizontal plane.
- Positive angles above horizontal plane.
- Map -2^31..2^31-1 to +/-90.
- Use -2^31 as an "out of range" indicator.
- Res: ~42 nano deg.

Conversion Formula:

\[
US = \text{(double)} \left( \frac{180}{0xFFFFFFFF} \times LS \right)
\]

\[
LS_{90\_dec} = \left( \frac{180}{4294967294} \times LS_{90} \right)
\]

Example Value:

-0.4315251 Degrees

Example LS Packet:

[K][L][V] = [0d90][0d2][0xFD 3D]

US Key

<table>
<thead>
<tr>
<th>US Name</th>
<th>06 0E 2B 34 01 01 01 07</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESD Digraph</td>
<td>07 01 10 01 05 00 00 00</td>
</tr>
</tbody>
</table>

ESD Name

<table>
<thead>
<tr>
<th>ESD Name</th>
<th>UAV Pitch (INS)</th>
</tr>
</thead>
</table>

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);

US Conversion

\[
US_{\_dec} = \left( \frac{180}{4294967294} \times LS_{\_int} \right)
\]

To US:
- US = (double)\(180/0xFFFFFFFF\times LS\)

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

ESD Conversion

\[
ESD_{\_dec} = \left( \frac{180}{4294967294} \times LS_{\_int} \right)
\]

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).
ST 0601.8 UAS Datalink Local Set

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>91</td>
<td>Platform Roll Angle (Full)</td>
<td>Degrees</td>
<td>+/- 90</td>
<td>int32</td>
</tr>
</tbody>
</table>

**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**

\[ LS_{\text{dec}} = \left( \frac{\text{LS\_range}}{4294967294} \right) \times \text{LS\_int} \]

\[ LS_{91\_\text{dec}} = \left( \frac{180}{4294967294} \right) \times \text{LS\_91} \]

**Example Value**
3.405814 Degrees

**Example LS Packet**
[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

**US Name**
Platform Roll Angle

**ESD Digraph**
UAV Roll (INS)

**US Conversion**

\[ \text{US\_dec} = \left( \frac{180}{4294967294} \right) \times \text{LS\_int} \]

**ESD Conversion**

\[ \text{ESD\_dec} = \left( \frac{180}{4294967294} \right) \times \text{LS\_int} \]

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag 92</td>
<td>Platform Angle of Attack</td>
<td>Degrees</td>
<td>+/- 90</td>
<td>int32</td>
</tr>
<tr>
<td></td>
<td>(Full)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>US</td>
<td>06 0E 2B 34 01 01 01 01</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mapped</td>
<td>0E 01 01 01 02 00 00 00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes:
- 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.

Conversion Formula:
\[
\begin{align*}
LS_{\text{dec}} &= \frac{LS_{\text{range}} \times LS_{\text{int}}}{180} \\
LS_{92\_\text{dec}} &= \frac{180}{2^{31}} \times \frac{LS_{92}}{2^{31}} - 1
\end{align*}
\]

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](image)

Note that the int32 used in the LS value is encoded using two’s complement.
8.93 Tag 93: Platform Sideslip Angle (Full) Conversion

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>Tag</th>
<th>LS Name</th>
<th>Notes</th>
</tr>
</thead>
</table>
| 93     |     | Platform Sideslip Angle (Full) | - 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) = 0x800000000.  
|        |     |                                  | - Res: ~42 nano deg.  

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degrees</td>
<td>+/- 180</td>
<td>int32</td>
</tr>
</tbody>
</table>

Notes:
- 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) = 0x800000000.
- Res: ~42 nano deg.

Conversion Formula:
- \[ \text{LS}\_\text{dec} = \left( \text{LS}\_\text{range} \times \text{LS}\_\text{int} \right) \]
- \[ \text{LS}\_93\_\text{dec} = \frac{360}{2^{31}-1 \times 2^{31}-1} \times \text{LS}\_93 \]

Example Value:
- \([K][L][V] = [0d93][0d]\)

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](image)

Note that the int32 used in the LS value is encoded using two’s complement.
8.94 Tag 94: MIIS Core Identifier

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>LS Name</th>
<th>US Mapped Key</th>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>94</td>
<td>MIIS Core Identifier</td>
<td>Use ST1204 MIIS Core 16-byte Key.</td>
<td>None</td>
<td>None</td>
<td>Binary Value</td>
</tr>
</tbody>
</table>

Notes
- Local set tag to include the ST1204 MIIS Core Identifier Binary Value within ST0601. Use according to the rules and requirements defined in ST1204.

Conversion Formula

Example Value

Example LS Packet

US Key

06 0E 2B 34 01 01 01 01
OE 01 04 05 03 00 00 00
Motion Imagery Identification System Core

ESD Digraph

x

ESD Name

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

<table>
<thead>
<tr>
<th>LS Tag</th>
<th>95</th>
<th>SAR Motion Imagery Metadata Use ST1206 SARMI 16-byte Key.</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>US Mapped Key</td>
<td></td>
<td></td>
</tr>
<tr>
<td>US</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Notes</th>
<th>Conversion Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>- 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.</td>
<td>x</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example Value</th>
<th>Example LS Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>x (R) (L) (V) = [0d 95 0d x]</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>US Key</th>
<th>06 0E 2B 34 02 0B 01 01</th>
</tr>
</thead>
<tbody>
<tr>
<td>US Name</td>
<td>0E 01 03 03 0D 00 00 00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Units</th>
<th>Range</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>None</td>
<td>Set</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ESD Name</th>
<th>SAR Motion Imagery Metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td></td>
</tr>
</tbody>
</table>

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 coherent 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].