Date : 2023.7.27

3D LiDAR
YLM-X001

COMMUNICATION SPECIFICATIONS
FOR DATA ACQUISITION

Symbol	Amended Reason	Pages	Date	Corrector	Amendment No.

Approved by	Checked by	Drawn by	Designed by	Title	YLM series Communication Specifications for data acquisition	
KAMITANI	HOSODA	ARIMORI	ARIMORI	Drawing No.	C-42-04549	1/14

Index

1. Overview …… 3
1.1 Term …… 3
2. Communication …… 4
2.1 Communication interface …… 4
2.2 Transmission of measurement data …… 4
3. Scanning pattern …… 5
4. (U, V) coordinate space …… 6
5. Packet structure …… 7
5.1 Packet structure details …… 7
5.2 Packet header …… 7
5.3 Global header …… 8
5.4 Type C …… 8
5.4.1 Type C header …… 8
5.4.2 Type C payload …… 10
5.5 Type D …… 11
5.5.1 Type D header …… 11
5.5.2 Type D payload …… 13
6. References …… 14

1. Overview

This is the communication specifications for data acquisition. The communication protocols described in this document are for point cloud data transmission and differ from the communication protocols for sensor setting. For the communication protocols for sensor setting, refer to the communication specifications for sensor setting.

1.1 Term

The definitions of terms described in this document is as follows.

Term	Definition
Virtualized sensor	Multiple FOV can be set in 1 unit and data for each FOV can be individually retrieved. That is, one sensor can be regarded as multiple sensors. These multiple sensors are each called “virtualized sensor”.
Base port number	1^{st} port number of virtualized sensor. In 2^{nd} and subsequent virtualized sensors, the port number is assigned in one increments from the base port number.
Payload	Group of data. The type of data varies depending on packet Type. Refer to 5. Packet structure for detailed information.
Mechanical-less	Structure with no mechanical components.
LCM	Abbreviation for Light Control Metasurface. Liquid crystal elements capable of reflecting incident light at arbitrary reflection angles.
Steer	Laser scanning direction by LCM. Also called as steering direction.
VCSEL	Element that is capable of emitting linear laser beam. *VCSEL in this document means 1D VCSEL array.
Stare	Spreading direction of VCSEL. Also called as non-steering direction.
θ, φ	Each corresponds to the angle in the steering direction and the angle in the non-steering direction.
Mapping table	Table containing pairs of (θ, φ) used when converting distance values to rectangular coordinates. Determined from the respective data included in the payload which θ and φ in the table correspond to the received distance value.
Sampling grid	Grid in (U, V) space that stores the measurement data.
Frame	Group of data scanned from the top edge of FOV to the bottom edge of FOV.
Binning	Process of grouping several pixels together to form a single pixel. The higher the binning level is, the lower resolution as the total number of pixels apparently decreases, but this condition improves sensitivity, accuracy and precision.

2. Communication

2.1 Communication interface

The communication interface of this sensor is Ethernet.

●Ethernet 1000BASE-TX

Communicated by TCP/IP. The default setting of network address is as follows;

IP address: 192.168.0.10
Subnet mask: 255.255.255.0
Base port number: 10940
*The port number of the virtualized sensor is incremented by 1 from the base port number. (10940, 10941, …10947)

2.2 Transmission of measurement data

This sensor starts scanning by sending the “start scan” command described in the communication specifications for sensor setting. During scanning, the sensor successively keeps sending the measurement data.
If there is not enough space in the receiving buffer to receive the measurement data, the measurement data is discarded. Note that the discarded data is not resent at this time.

3. Scanning pattern

The following figure shows the scanning pattern of this sensor. Multiple FOV can be set in vertical direction in this sensor (up to 8 patterns). Besides, LCM (Light Control Metasurface) enables the discrete measurement in vertical direction.
The figure below indicates a case of setting 2 FOV: FOV0 for upper FOV and FOV1 for bottom FOV. This sensor measures set FOV only. In addition, FOV0 and FOV1 are alternately measured one line for each. This scanning order can be changed by “interleave” described in the communication specifications for sensor setting.

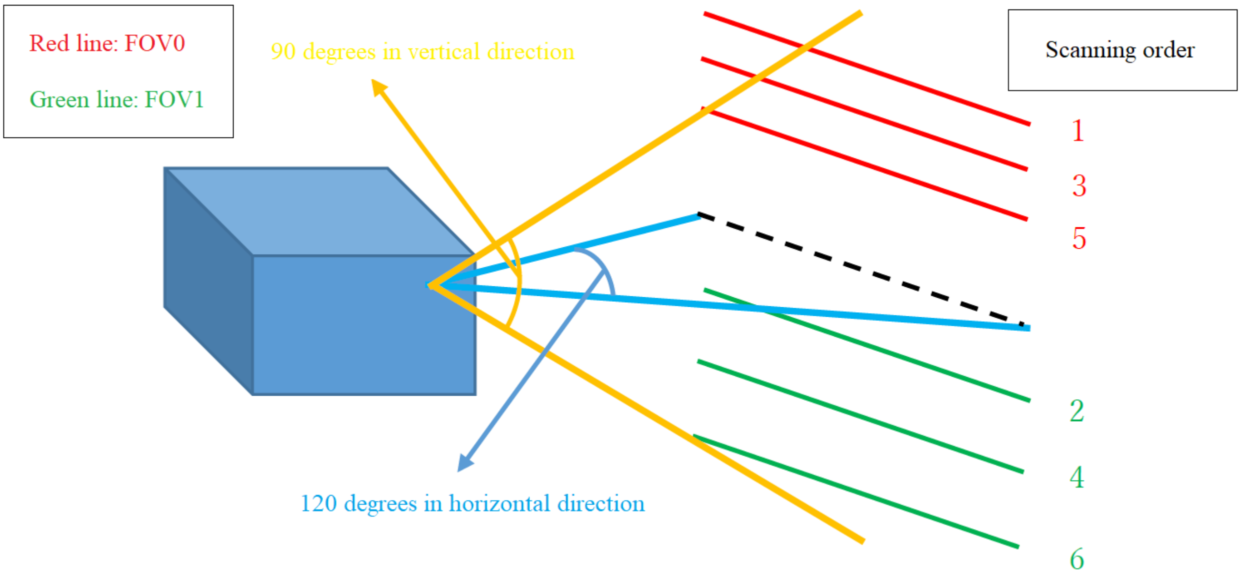

Figure 1 Example of scanning pattern

4. (U, V) coordinate space

This chapter describes (U, V) coordinate space. (U, V) coordinate space means the spherical coordinate space consisting of U dimension in the non-steering direction and V dimension in the steering direction. The conversion from (U, V) coordinate space to rectangular coordinates is performed with the mapping table provided by Type C packet.

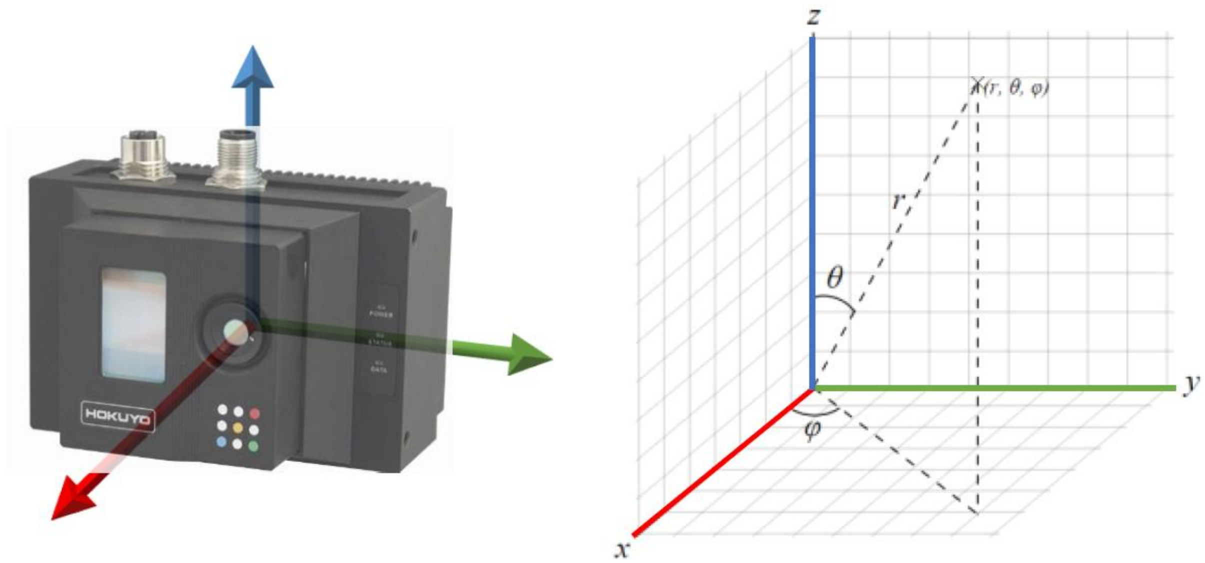

Figure 2 Polar coodinates and rectangular coodinates

The following formulas show how to convert (U, V) coordinates to rectangular coordinates with (φ, θ) of the mapping table.

$$x=r\cos\varphi\sin\theta$$
$$y=r\sin\varphi\cos\theta$$
$$z=r\cos\theta$$

The apparent number of pixels can be changed by binning. When binning is carried out, the center position of the pixel after binning is between the pixel and the pixel before binning. Therefore, if the resolution of the mapping table is same as the resolution of the sensor, pairs of φ and θ used in the binning pixel will be slightly misaligned from the center of the pixel. In order to reduce the misalignment, the resolution of the mapping table is set higher than the resolution of the sensor.

The mapping table is sent as Type C packet when starting scan, when TCP connection is established and regularly. The mapping table should be cached at the receiving end in advance and used for the coordinate conversion of the measurement data transmitted in Type D packet. The mapping table contains pairs of φ and θ for all ranges that may be measured.

The sampling grid of each virtualized sensor are sized to correspond to FOV set for each sensor. The specific size is described in Type D header. The sampling grid of each virtualized sensor can be created by using the size and (Steer, Stare) offset of Type D header. The corresponding mapping table can be referred to by using the (U, V) offset and step in Type D header. Steps other than 1 represent the gap between the mapping table and the sampling grid.

Here are the image figures of the mapping table and the sampling grid.
FOV: -45~45 degress in vertical direction
Binning: without binning

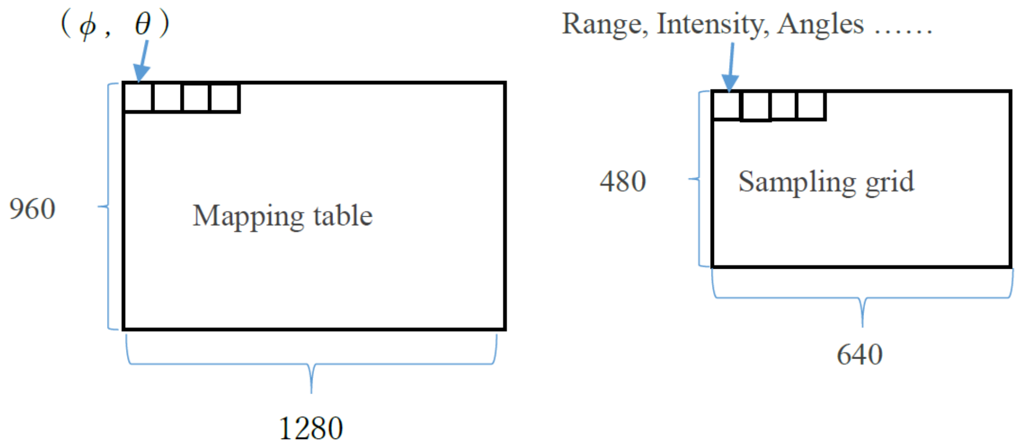

Figure 3 Mapping table and sampling grid

5. Packet structure

All packets consist of 4 parts and connected without padding. Unless otherwise described, the data is binary format and the byte order is big endian.

1. Packet header
2. Global header
3. Header by type
4. Payload by type

Which Type the received packet is read from the global header. There are two kinds of Types: Type C and Type D. Each contains the mapping table and the measurement data. Up to 64 elements of data are included in 1 payload.

Packet header
Global header
Header by type
Payload by type

Figure 4 Packet whole structure

5.1 Packet structure details

The structure of each header/payload is shown from the following section. In each section, the structure of the header/payload is firstly described in the figure and the description of each item is given next. The numbers in the upper part of each structural figure are in bits.

The lower part of the structural figure is described as follows;

Item (first bit, last bit, length): description

"Item": the name of each item on the structural figure.

"first bit"/"last bit": the first and last bits of the area in which the item is located, with the top left corner of the structural figure as bit 0.

"length": the data length of the item in bits.

“description": the description of the item.

5.2 Packet header

The structure of packet header is described below. The length of whole packet is included in the packet header.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Length																															
Flags																															
Flag Dependent Value1																															
Flag Dependent Value2																															

Figure 5 Packet header structure

Length (0, 31, 32): Unsigned integer, in octets. The length of the framed packet data (not inclusive of this 4-byte header). Can be zero, either to push data in the flags and value fields, or as a connection keepalive/heartbeat.

Flags (32, 63, 32): Bit field. All: Reserved

Flag Dependent Value 1 (64, 95, 32): Dependent on flags value. May encode optional framing-layer data when flags indicate.

Flag Dependent Value 2(96, 127, 32): Dependent on flags value. May encode optional framing-layer data when flags indicate.

5.3 Global header

The structure of global header is described below. The data on the identification of the sensor and the type of packet are included in the global header.

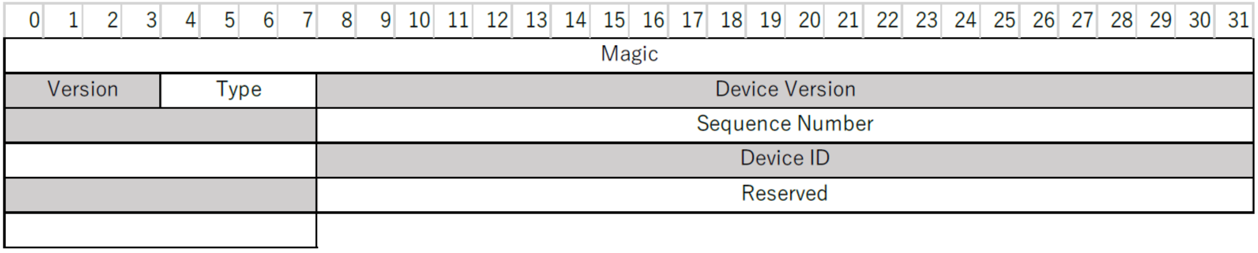

Figure 6 Global header structure

Magic(0, 31, 32): ASCII without null-termination. Always equal to 'BCDA'. This serves to identify the protocol in the case of port conflicts.

Version(32, 35, 4): Unsigned integer. Equal to 1.

Type(36, 39, 4): Unsigned integer enum. Payload type, identifies the payload header that is to follow. Currently two types are defined:

Value	Description
0×C	Indicates that it is Type C packet; Type C packet contains the mapping table.
0×D	Indicates that it is Type D packet; Type D packet contains the measurement data.

Device Version(40, 71, 32): Not implemented at this moment. The firmware version will be implemented for the future.

Sequence Number(72, 103, 32): Unsigned integer. Arbitrary starting point (when receiver joins or connects). Wraps around to zero.

Device ID(104, 135, 32): Not implemented at this moment. The device ID will be implemented for the future. The device ID is an identifier set in the sensor during start-up calibration.

Reserved(136, 167, 32): Reserved for future use. May be further subdivided later.

5.4 Type C

This section describes Type C packet. The mapping table to convert (U, V) spherical coordinates to rectangular coordinates is contained in Type C packet.

5.4.1 Type C header

The structure of Type C header is described below.

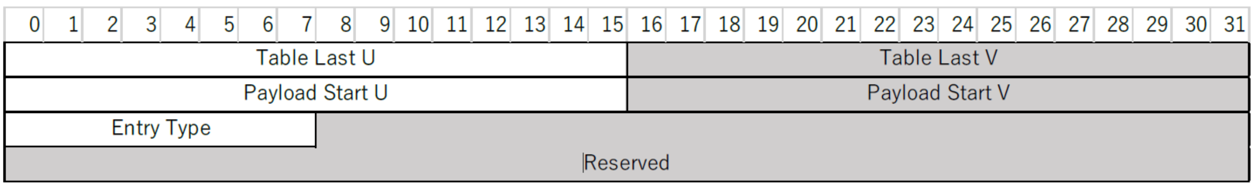

Figure 7 Structure of Type C header

Table Last U(0, 15, 16): Unsigned integer. Size of the mapping table in "U" dimension. This number defines the size of the logical sampling grid on which the payload returns exist. It is recommended that any fixed-sized structures in the receiver have at least size (U+1) in the "U" dimension.

Table Last V(16, 31, 16): Unsigned integer. Size of the mapping table in "V" dimension. This number defines the size of the logical sampling grid on which the payload returns exist. It is recommended that any fixed-sized structures in the receiver have at least size (V+1) in the "V" dimension.

Payload Start U(32, 47, 16): Unsigned integer. The U index of the first element in this packet.

Payload Start V(48, 63, 16): Unsigned integer. The V index of the first element in this packet.

Entry Type(64, 71, 8): Unsigned integer. Defines type of the mapping table elements that follow:

Value	Description
2	Indicates that the payload data is a pair of θ and φ.

Reserved(72, 127, 56): Reserved for future use.

5.4.2 Type C payload

The Type C packet payload when Entry Type is 2. The first pair of θ and φ in this payload is the (φ, θ) in the (U, V) index specified in Type C header. From the second element, the pairs of θ and φ in the U-dimensional direction are stored. Within the same packet, data does not wrap to the next line.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Theta																															
Phi																															
Next Entry…																															

Figure 8 Structure of Type C payload

Theta(0, 31, 32): Two's complement signed integer, arcsecond units. The spherical coordinate theta to which this (U, V) entry should map. Maximum and minimum values (2^{31}-1, -2^{31}) indicate mapping entry not present.

Phi(32, 63, 32): Two's complement signed integer, arcsecond units. The spherical coordinate phi to which this (U, V) entry should map. Maximum and minimum values (2^{31}-1, -2^{31}) indicate mapping entry not present.

5.5 Type D

This section describes Type D packet, which contains the measurement data output from each virtualized sensor.

5.5.1 Type D header

The structure of Type D header is described below.

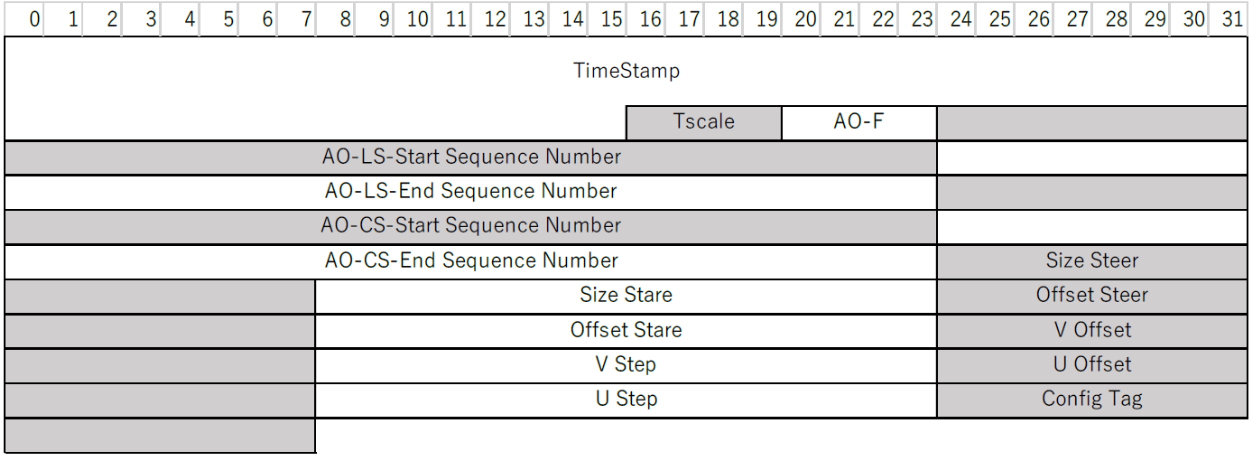

Figure 9 Structure of Type D header

Timestamp(0, 79, 80): PTP format timestamp (48 bit-seconds || 32 bit-nanoseconds) based on TScale. The time at software initialization is set to 0. The timestamp is updated every 80 packets.
In software version R5.0.2, the timestamp update timing is frame by frame.

TScale(80, 83, 4): Unsigned integer enum. Defines timescale that's in use for timestamping:

Value	Description
0	TAI
1	UTC
2	GPS
3	ARB (Arbitrary, assumed to be the same arbitrary source with which the receiver(s) are synchronized. Corresponds with PTP's "ARB" semantics.)

AO-F(84, 87, 4): Flags for following "Advisory Only" Fields:

Value	Description
0×01	AO-LS-Start Sequence Number is Present and Valid.
0×02	AO-LS-End Sequence Number is Present and Valid.
0×04	AO-CS-Start Sequence Number is Present and Valid.
0×08	AO-CS-End Sequence Number is Present and Valid.

AO-LS-Start Sequence Number(88, 119, 32): Unsigned integer. Sequence number of the first packet of the previous scene.

AO-LS-End Sequence Number(120, 151, 32): Unsigned integer. Sequence number of the final packet of the previous scene.

AO-CS-Start Sequence Number(152, 183, 32): Unsigned integer. Sequence number of the first packet of the current scene. May be equal to the current packet's sequence number, if this is the first packet in a scene.

AO-CS-End Sequence Number(184, 215, 32): Unsigned integer. Sequence number of the last packet of the current scene. May be equal to the current packet's sequence number, if this is the last packet in a scene. Will often be unset until the last packet in a scene, if at all. If never set or packet is dropped, can be recovered by looking at "A-LS-**" in subsequent packets.

Size Steer(216, 231, 16): Unsigned integer. The size of the scene in the "steering" dimension of the sensor. Note that the entire scene may not be sampled during a particular scan, or ever. This number defines the size of the logical sampling grid on which the payload returns exist if the (U,V) space is not used by the client. It is recommended that any fixed-sized structures in the receiver have at least this size in the steering dimension.

Size Stare(232, 247, 16): Unsigned integer. The size of the scene in the "non-steering" dimension of the sensor. Note that the entire scene may not be sampled during a particular scan, or ever. This number defines the size of the logical sampling grid on which the payload returns exist if the (U,V) space is not used by the client. It is recommended that any fixed-sized structures in the receiver have at least this size.

Offset Steer(248, 263, 16): Unsigned integer. The zero-based steering-dimension offset at which the payload samples are to be placed in the scene if the (U,V) space is not used by the client.

Offset Stare(264, 279, 16): Unsigned integer. The zero-based staring-dimension offset at which the payload samples are to be placed in the scene if the (U,V) space is not used by the client.

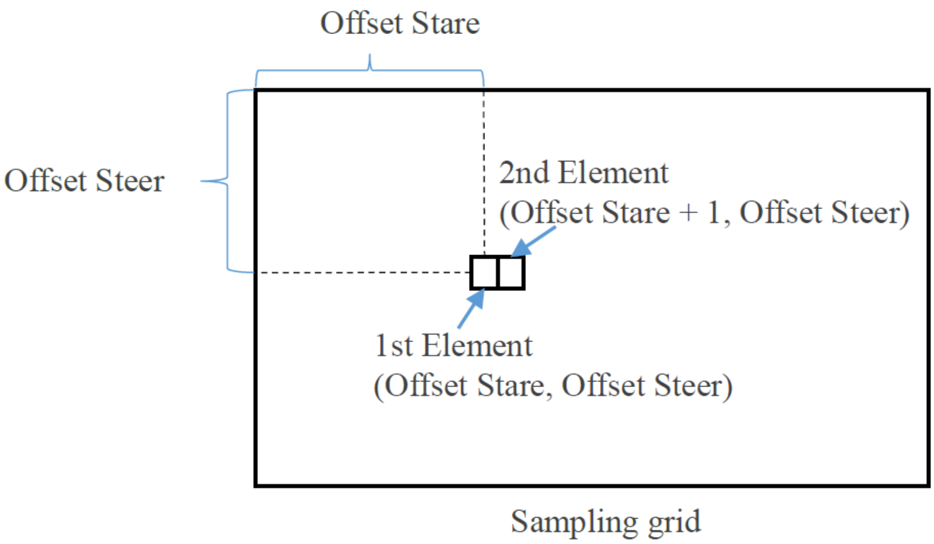

Figure 10 Relationship between (Offset stare, Offset Steer) and sampling grid

V Offset(280, 295, 16): Unsigned integer. Indicates which V index in the mapping table corresponds to the first element of the payload of this packet.

V Step(296, 311, 16): Unsigned integer. Determines how much the index of the V dimension of the mapping table is increased when moving to the next element of the payload. This value is not used for accessing the mapping table because only the U index is incremented in the same packet.

U Offset(312, 327, 16): Unsigned integer. Indicates which U-index in the mapping table corresponds to the first element of the payload of this packet.

U Step(328, 347, 16): Unsigned integer. Determines how much the index of the U dimension of the mapping table is incremented when moving to the next .

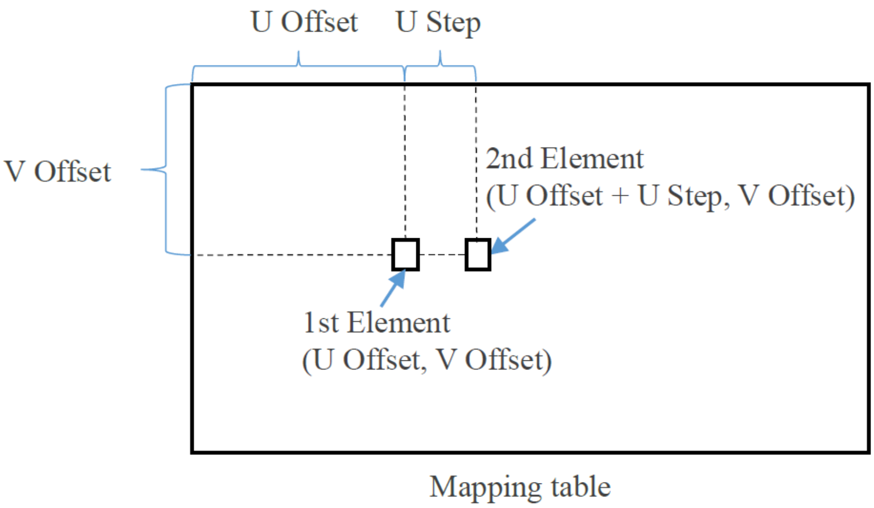

Figure 11　Relationship between (U Offset, V Offset), U Step and mapping table

Config Tag(348, 363, 16): Opaque identifier (may be LSB justified if less than 16 bits are present). The tag value of the sensor configuration which was used to generate the payload. This corresponds to “user_tag” described in the communication specifications for sensor setting.

5.5.2 Type D payload

Type D packet payload. This uses sizes and offsets in the non-steering and steering directions to store the measurement data in the sampling grid of the virtualized sensor. “Stare Offset” indicates U index of the first element of the payload and “Steer Offset” indicates V index. The index of subsequent measurement data increases in the non-steering dimension. Packets may be padded with invalid measurement data or zeros. However, the line will not be folded beyond the size of the non-steering dimension.
To refer to the mapping table corresponding to these elements, use (U Offset, V Offset) and U Step. The first element of the payload corresponds to (U Offset, V Offset) in the mapping table. In addition, the mapping table to which the following elements correspond is (U Offset + U Step, V Offset). Note that the V Step is not used to read out data, but can be used to estimate the degree of subsampling in the steering direction.

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td colspan="16">Intensity</td><td colspan="16">Range</td></tr>
<tr><td colspan="16">Background</td><td colspan="16">SNR</td></tr>
<tr><td colspan="8">Reserved</td><td colspan="8">Flags</td><td colspan="16"></td></tr>
<tr><td colspan="32">Next Returns…</td></tr>
</table>

Figure 12　Structure of Type D payload

Intensity(0, 15, 16): Unsigned integer, arbitrary linear unit. The intensity of a return. May not be present if none identified or filtering criteria not met at corresponding scene point. MUST be ignored if the corresponding present and valid flag is not set.

Range(16, 31, 16): Unsigned integer, arbitrary linear unit. The range to a return. May not be present if none identified or filtering criteria not met at corresponding scene point. MUST be ignored if the corresponding present and valid flag is not set. The distance value consists of a 6-bit integer part and a 10-bit decimal part. Therefore, the distance value R[m], where M is the 16-bit integer value, is calculated by the following formula.

$$R = M \times \ 2^{-10}[m]$$

Background(32, 47, 16): Unsigned integer, arbitrary linear unit. The background level at the corresponding scene point. MUST be ignored if the corresponding present and valid flag is not set.

SNR(48, 63, 16): Unsigned integer, arbitrary linear unit. The signal to noise ratio at the corresponding scene point. MUST be ignored if the corresponding present and valid flag is not set.

$$\mathrm{SNR} = \mathrm{K} \times \ 2^{-3}$$

Reserved(64, 71, 8): Reserved for future use.

Flags(72, 79, 8): Flags associated with this return:

Value	Description
0×01	Range Present and Valid
0×02	Intensity Present and Valid
0×04	Background Present and Valid
0×08	SNR Present and Valid

6. References

Title of document	Reference (Document number or URL)
YLM series Communication Specifications for sensor setting	C-42-04550