2019-1-24

Delta-2ACommunication interface protocol

[Delta-2A]

Shenzhen Shanchuan Robotics Co., Ltd.

Table of contents

1. Introduction to Radar Communications

Delta-2A LiDAR communicates with external devices via UART TTL level and only supports simplex

Communication (i.e. the laser radar actively sends data frames to external devices), and the external device only needs to extract valid

No response is needed. All data in the communication frame are in hexadecimal format.

The radar rotates and measures once, scanning and obtaining information about evenly distributed points around it (angle and distance of the points).

The SDK receives the parsed data and obtains the information of each circle point. A circle of 360° is evenly divided into 16

The frame reports the scanning information (see the command word list below) frame, so the starting angle of each frame of 16 frames is

It is 0° (zero point - see the specification for position), 22.5°, 45°, 67.5°, 90°...270°, 292.5°, 315,

337.5°, 360°. 16 frames of data add up to a complete circle, and the total number of points in a circle = 16*the number of points in each frame;

The total number of points per frame can be obtained by calculating the number of distances based on the scanned information frame (number of distances = total number of points).

Data point information (angle and distance): The distance of the Nth point in a frame is the Nth distance value in the scan information frame.

The angle corresponding to the distance of the Nth point in that frame = the starting angle of this frame + (N-1) * 22.5 / (the total angle of each frame

Points), so that all the point information (angle and distance) of one frame are available.

According to the communication protocol defined in this article, communication data can be parsed to obtain real-time measurement information and equipment

Health status information.

2. Communication frame structure

The communication frame consists of frame header, frame length, frame type, command word, parameter length, parameter, and check code.

It is mainly used for the laser radar to actively upload measurement information, fault information, etc. to the external host. The host only needs to

Just extract the valid data from the communication frame uploaded by the radar, no response is required.

The command frame format is as follows:

Frame Header	Frame length	Protocol Version	Frame Type	Command word	Parameter length	parameter	Check code

Frame header: The frame header field occupies 1 Byte and is fixed at 0xAA.

Frame length: The frame length field occupies 2 bytes. The frame length is calculated from the frame header to the byte before the check code.

The high position is in front and the low position is in the back.

Protocol Version: The address code field occupies 1 Byte and the default value is 0x00.

Frame type: The frame type field occupies 1 Byte and is fixed at 0x61.

Command word: The command word field occupies 1 Byte and is an identifier to distinguish different commands.

Parameter length: The parameter length occupies 2 bytes, which is the length of the valid data in the data frame, with the high bit in front and the low bit in the back.

parameter: The parameter field is the valid data for the command.

Verification code: The check code field is a 16-bit cumulative sum, occupying two bytes, with the high bit first and the low bit last.

calculate: The accumulated sum from the frame header to the byte before the check code.

Command word list:

Command word	describe	Parameter length	Parameter Description
0xAD	Measurement information	(3N+5)Bytes	0Bytes: Radar rotation speed value, 8 bits unsigned number, minimum resolution 0.05r/s (i.e. the speed value is 1, the corresponding speed is 0.05r/s) 1～2Bytes: zero offset, 16 bits signed number, high bit first, The minimum resolution is 0.01° (zero offset: radar adjustment Test information, not used after parsing) 3～4Bytes: The starting angle value of this data frame is a 16-bit unsigned number with the high bit first and the low bit first. 5 Bytes: Signal value corresponding to distance value 1, 8 bits unsigned number (signal

			Number value: radar debugging information, not used after parsing) 6～7Bytes: Distance value 1, 16 bits unsigned number, high bit first, low bit last 8Bytes: The signal value corresponding to the distance value 2, 8 bits unsigned number (signal value: Debug information, not used after parsing) 9～10Bytes: Distance value 2, 16 bits unsigned number, high bit first, low bit last ..... 3N + 2Bytes: The signal value corresponding to the distance value N, 8 bits unsigned Number (signal value: radar debugging information, not used after parsing) 3N + 3 ~ 3N + 4Bytes: Distance value N, 16 bits unsigned number, high bit first, low bit last Remark: 1. Angle range: 0 ~ 36000 2. Angular resolution: 0.01° (i.e., the angle value is 1, the corresponding angle is 0.01°) Distance resolution 0.25mm (i.e. distance value 1 corresponds to the actual distance is 0.25mm) 3. Angle calculation:

			Example: Distance n (n is 1 to N, N is the number of distance points in this frame) corresponding to the angle calculation: N = (parameter length - 5)/3 Angle at distance n = starting angle + 22.5°*(n - 1)/N
0xAE	Device Health information	1 Byte	Equipment speed failure Speed value, 8 bits unsigned number, minimum resolution is 0.05r/s

3. Checksum calculation

The communication frame check algorithm of this protocol adopts 16-bit cumulative sum. The following is a routine to calculate the check code.

For reference only.

```
//==============================================
================================

// Check code calculation

// *Start_Byte: start byte

// Num_Bytes: length of the calculated data

// Return value: 16-bit checksum

//==============================================
================================

u16 CRC16(u8 *Start_Byte,u16 Num_Bytes)

{
    u16 Checksum = 0;

    while (Num_Bytes--)

    {    // Calculate CRC

        Checksum += *Start_Byte++;

    }

    return Checksum;

}
```

4. Communication frame example analysis

0.**Resolution in protocol:**Actual measurement data = value in communication * resolution

Actual speed = speed value in communication * resolution (0.05r/s)

Actual distance = communication distance value * resolution (0.25mm)

Actual angle = angle value in communication * resolution (0.01°)

1. Measurement data frame:

```
AA 00 9A 01 61 AD 00 92 82 00 87 69 78 00 00 00 46 21 3A 54 23 78

00 00 00 00 00 00 91 33 60 82 32 F7 93 32 EB 6D 32 E0 51 21 88 00 00

00 5D 21 88 66 21 8D 68 21 BF 41 32 D4 86 33 02 4D 32 E0 89 51 48 8E

51 48 92 51 48 8C 51 48 63 50 19 6D 51 48 7C 51 64 92 51 64 89 51 48

90 51 64 89 51 48 93 51 64 4B 53 2D 57 59 BA 43 2F 78 41 2E E4 00 00
```

00 54 2E DE 6B 2E E4 6B 2F 50 58 2E E4 7E 2F 64 5D 2F 78 3F 5A 0B 5A

5B FD 57 5B D3 5B 5C 28 59 5C 28 59 5B FD 5E 5E 32 35 BC

AA: Frame Header

00 9A: The frame length is 0x009A (Note: it is only the frame length of the example frame, not the actual length of the radar)

01: Protocol Version

61: Frame Type

AD: Command word

00 92: Valid data length 0x0047

82: Radar speed 130*0.05r/s (resolution) = 6.50r/s

00 87: Zero offset

69 78: Starting angle 27000*0.01° (resolution) = 270°

00: Signal letter 1

00 00: Distance value 1 is 0*0.25mm (resolution) = 0mm

46: Signal value 2

21 3A: Distance value 2 is 8506*0.25mm (resolution) = 2126mm

....

5E: Signal value 47

5E 32: Distance value 47 is 24114*0.25mm (resolution) = 6028mm

35 BC: Check code 0x35BC=(AA+00+9A+...+FD+5E+5E+32)

2. Radar speed fault frame:

AA 00 09 00 61 AE 00 01 69 02 2C

AA: Frame header identifier.

00 09: The frame length is 0x0009 (i.e. 9) bytes (excluding CRC code)

00: Protocol Version

61: Frame Type

AE: Command word

00 01: Valid data length 0x0001

C9: Radar speed 0xC9, i.e. 201*0.05r/s (resolution) = 10.05r/s

02 2C: Check code 0x022c