SLAMTEC

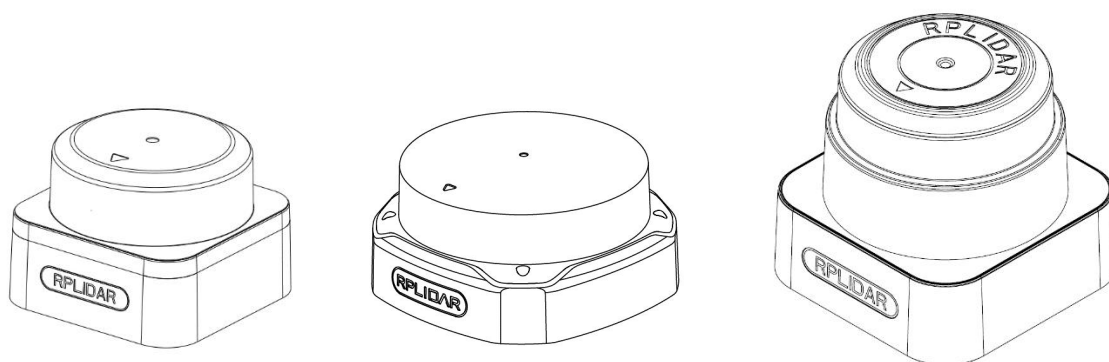# RPLIDAR
360 Degree Laser Range Scanner
Interface Protocol and Application Notes
Applied to RPLIDAR S Series

Contents

The host system communicates with RPLIDAR core system via the TTL UART serial interface. Based on the communication protocol defined in this document, the host system can retrieve the scan data, the device status, the health information etc. and manipulate RPLIDAR's working mode.
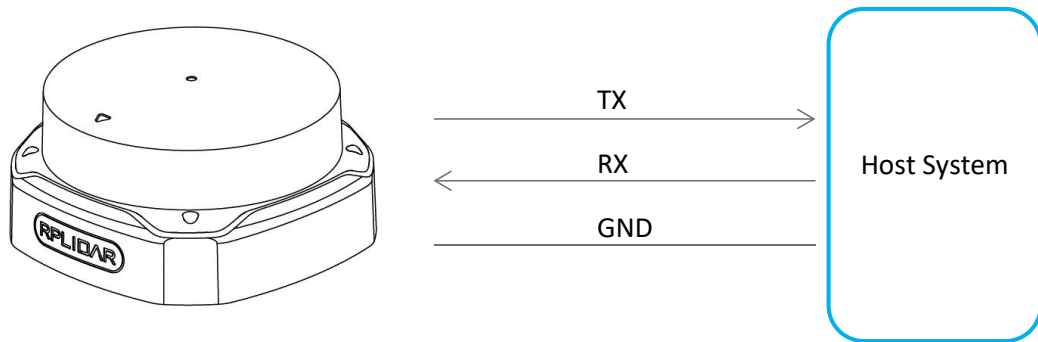

Figure 1-1 The Communication Between RPLIDAR and Host System

Please refer to the RPLIDAR datasheet for information about the bottom layer communication protocol and the electrical level definition of the serial signals used to communicate with RPLIDAR. The communication protocol based on UART serial port and the data transmission format will be introduced in this document.

SDK and Demo Program

SLAMTEC provides open-sourced SDK and demo program for customers to integrate RPLIDAR into their systems quickly. The SDK implements all the communication stacks, driver logics and related data structures described in this document. The public SDK is open-sourced and can be retrieved from GitHub: https://github.com/slamtec/rplidar_sdk .

The SDK supports multiple platforms including Windows, Linux, MacOS and even bare systems without an OS.

Please refer to the SDK manual for details.

Basic Communication Mode

The RPLIDAR uses a non-textual binary data packet based protocol to communicate with host systems. And all the packets transmitted on the interface channel share uniform packet formats.

A communication session is always initialized by a host system, i.e. a MCU, a PC, etc. RPLIDAR itself won't send any data out automatically after powering up.

If a data packet is sent from host systems to RPLIDARs, such a packet is called a **Request.** Once an RPLIAR receives a request, it will reply the host system with a data packet called a **Response**.

RPLIDAR will only start performing related operations required by a host system once after it receives a request. If RPLIDAR should reply to the host system, it will send one or more required response packets.

In order to let an RPLIDAR start scanning operation and send out data, a host system is required to send a pre-defined **Start Scan** request packet to RPLIDAR. RPLIDAR will start scanning operation once after it receives the request and the scan result data is sent out to the host system continuously.

There are three different request/response modes based on the related request types:
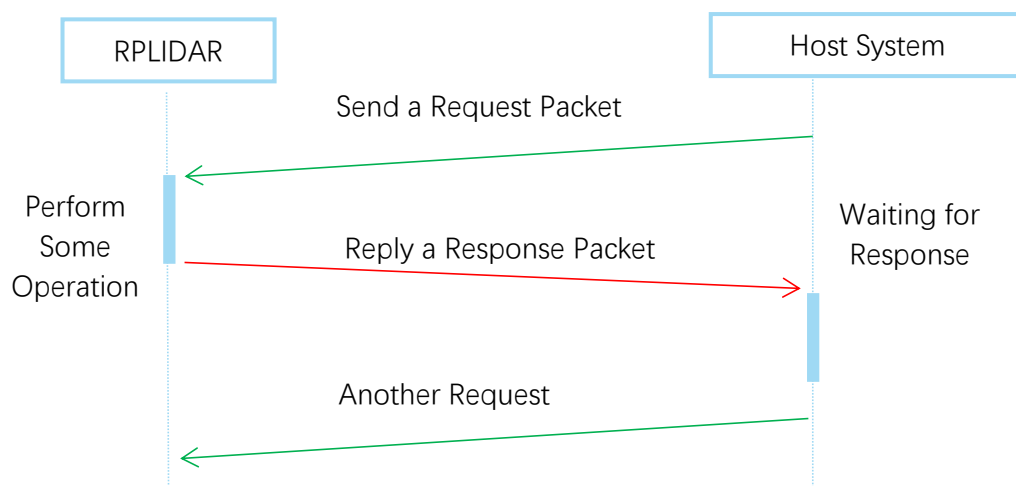
Request/Response Modes

Figure 2-1 RPLIDAR Request/Response Modes

==Host system should prevent sending extra request packets if the RPLIDAR is busy with handing the current request and hasn't replied to the host system yet. Otherwise,==

these extra request packets will be discarded by the RPLIDAR's protocol stack and the RPLIDAR will not have any chance to handle it.

Single Request-Multiple Response Modes

This mode is used when the RPLIAR is asked to perform the scan operation. After a host system sending a Start Scan request, RPLIDAR will take distance scan measurement continuously. Once a scan measurement sample is retrieved, its related result data (distance, angle value) will be sent out as individual response packets.

Host systems are only required to send a single request packet with this mode but will receive a continuous response packet stream with multiple response packets.
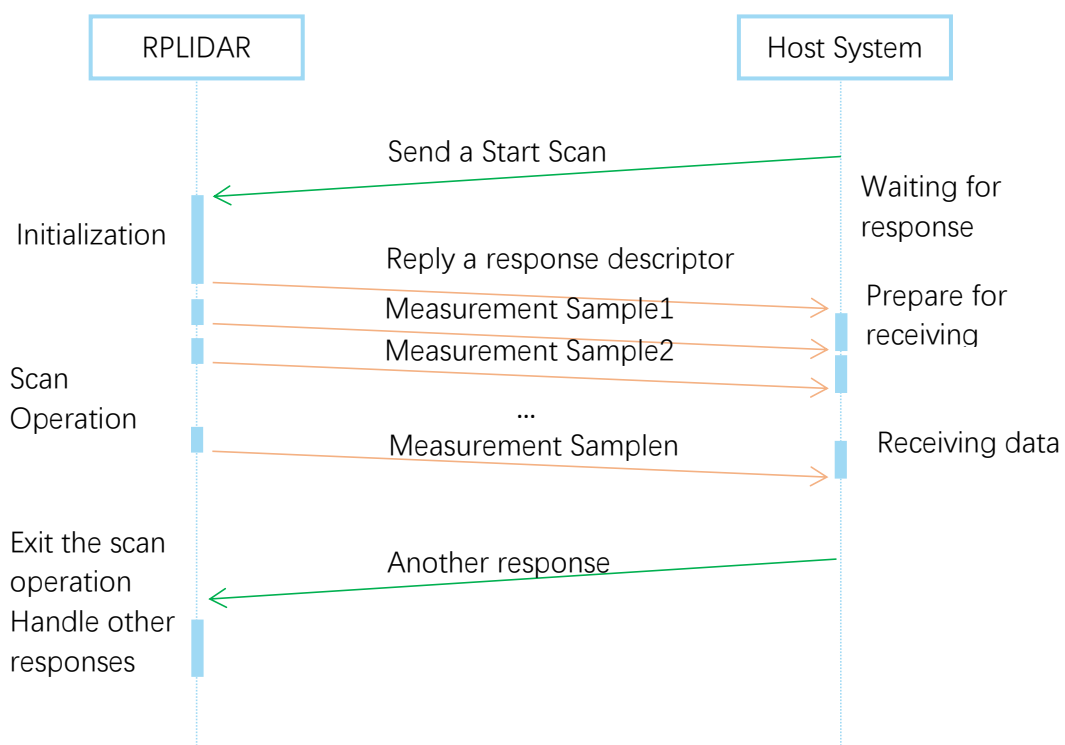

Figure 2-2 RPLIDAR Single Request - Multiple Response Mode

The host system can interrupt RPLIDAR and let it leave the multiple responses mode by sending a STOP request or any request packet. After leaving the multiple

responses mode, the RPLIDAR will continue to handle the request which has interrupted it.

The request packets sent by the host system during the multiple responses mode will be cached by the RPLIDAR's protocol stack. After leaving the multiple responses mode, RPLIDAR will handle the cached request.

Single Request-No Response

For requests like STOP, RESET Core, RPLIDAR uses the single request – no response mode since there is no need to reply to the host system.

Host systems should wait for a period of time before sending another request since RPLIDAR needs time to process the request operation. Otherwise, the request may be discarded by the RPLIDAR's protocol stack.
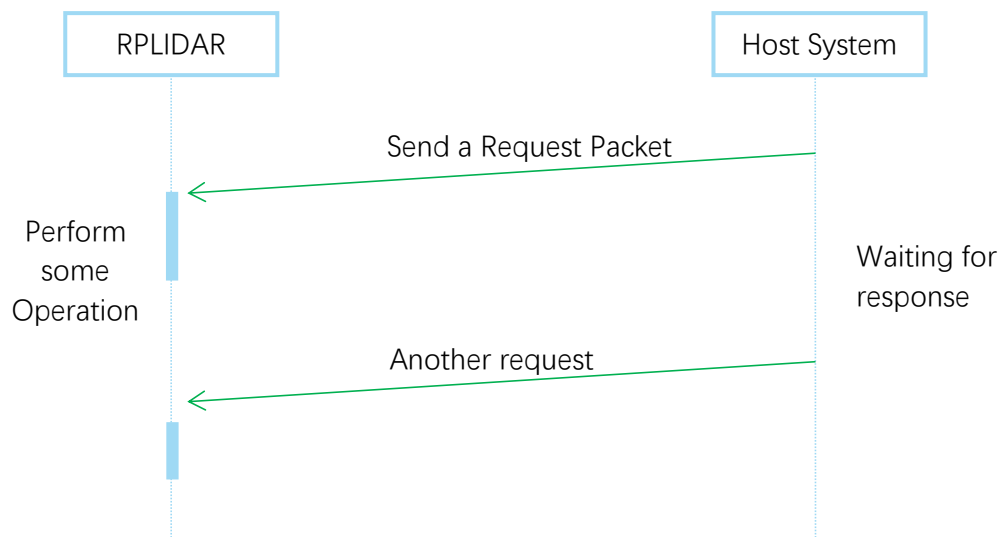
Figure 2-3 RPLIDAR Single Request-No Response Mode

Request Packets' Format

All request packets sent by a host system share the following common format. Little endian byte order is used.

Start Flag	Command	Payload Size	Payload Data	Checksum
1byte	1byte	1byte	0-255 bytes	1byte

Transmission Order →   Optional Section   ≤5 seconds

Figure 2-4 RPLIDAR Request Packets' Format

A fixed 0xA5 byte is used for each request packet, RPLIDAR uses this byte as the identification of a new request packet. An 8bit (1byte) command field must follow the start flag byte.
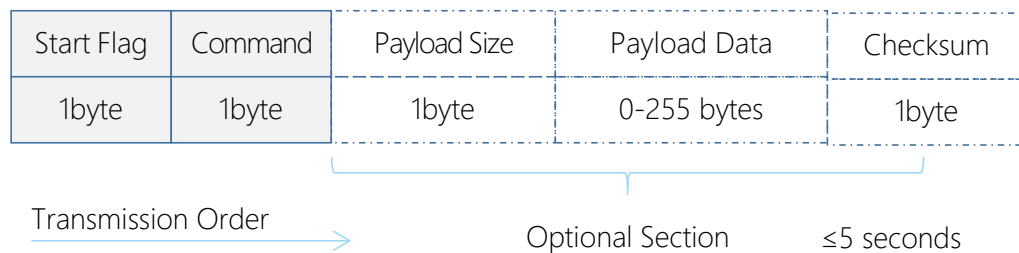
If the current request carries extra payload data, an 8bit (1byte) payload size field is required to be transmitted after sending the command field and then follows the payload data. After the payload data has been transmitted, an 8bit (1byte) checksum field calculated from the previous sent data should be transmitted.

The checksum value can be calculated using the following equation:

$$checksum = 0 \oplus 0xA5 \oplus CmdType \oplus PayloadSize \oplus Payload[0] \oplus ... \oplus Payload[n]$$

Note: Timing Consideration

All bytes within a request packet must be transmitted to RPLIDAR within 5 seconds. Otherwise, the communication stack of RPLIDAR will discard the current request packet.

Response Packets' Format

All the response packets are divided into two classes: **response descriptors** and **data responses**. If the current request received by RPLIDAR requires a response, RPLDAR will always send a response descriptor packet first and then send one or more data response packets based on the type of requests. Only one response descriptor packet will be sent out during a request/response session. The response descriptors carry the information of the incoming data responses. All the response descriptors share a same format.
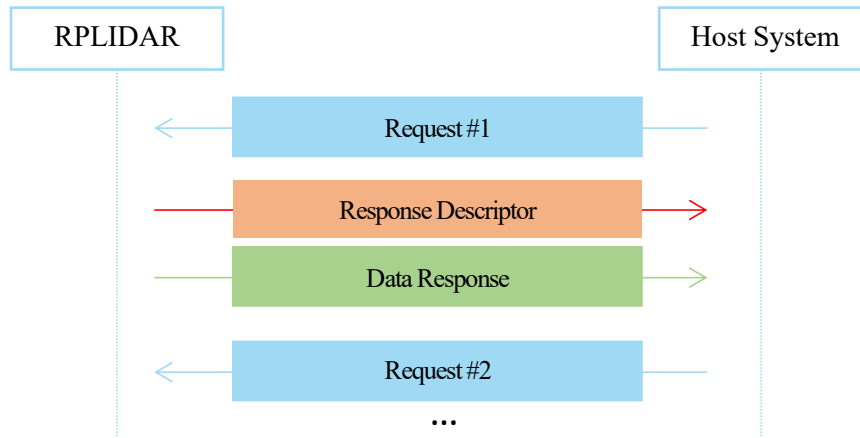
Figure 2-5 Response Packets Sent during a Single Request-Single Response Mode

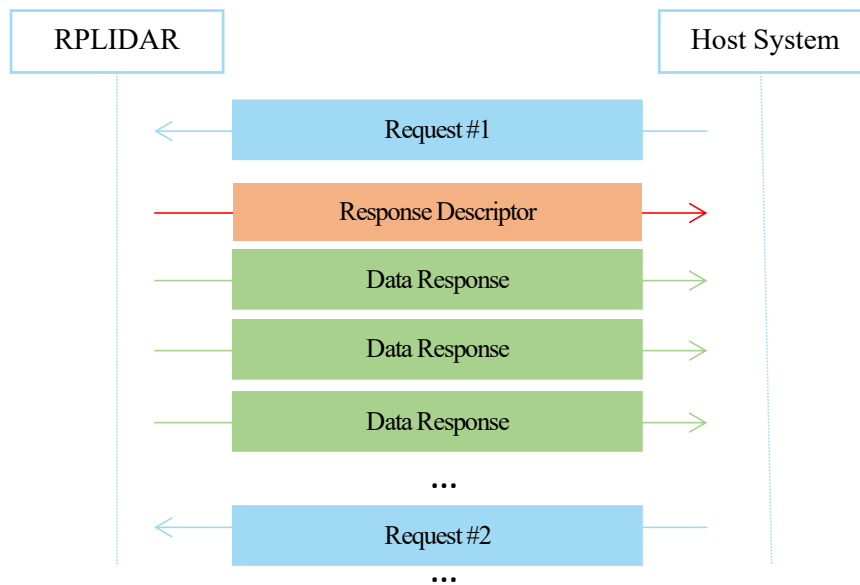*Figure 2-6 Response Packets Sent during a Single Request-Multiple Response Mode*

The format of response descriptors is depicted in the following figure.

Start Flag1	Start Flag2	Data Response Length	Send Mode	Data Type
1byte (0xA5)	1byte (0x5A)	30bits	2bits	1byte

Transmission Order

Figure 2-7 RPLIDAR Response Descriptors' Format

A response descriptor uses fixed two bytes' pattern 0xA5 0x5A for the host system to identify the start of a response descriptor. The 30bit Data Response Length field records the size of a **single** incoming data response packet in bytes. (All the incoming data response packets within a request/response session should have the same format and length). The 2bits Send Mode field describes the request/response mode of the current session. Its values are listed below:

Send Mode	Description
0x0	Single Request – Single Response mode, RPLIDAR will send only one data response packet in the current session.
0x1	Single Request – Multiple Response mode, RPLIDAR will continuously send out data response packets with the same format in the current session.
0x2	Reserved for future use
0x3	Reserved for future use

Figure 2-8 RPLIDAR Data Response Packets Value

The 1byte Data Type describes the type of the incoming data response packets. It is related to the type of the request RPLIDAR just received. Host systems can choose different data receiving and handling policy based on this field.

Different from response descriptors, there is no common format used among response data packets. Each type of response data has its own data format and packet length based on its type.

Major Working States and Transition Conditions

RPLIDAR has the following 4 major states: Idle, Scanning, Request Processing and the Protection Stop state.

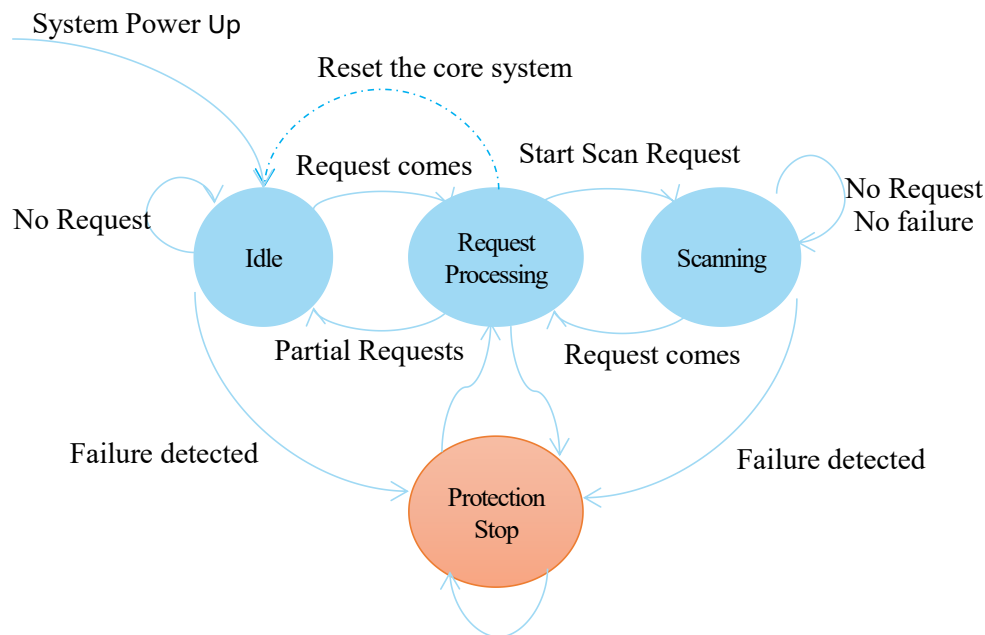The translation conditions are depicted in the following figure:

Figure 3-1 RPLIDAR's Major Status Translation

The Idle state is the default state of RPLIDAR which will be entered automatically after powering up or reset. Both the laser diode and the measurement system are disabled in this state, and the whole system is in power saving mode. Once RPLIDAR enters the Scanning state, the laser diode and the measurement system will be enabled and RPLIDAR will start measuring distance and sending the result out continuously.

RPLIDAR will enter the Request Processing state once it receives request packets from the host system. During processing the request, RPLIDAR won't perform the scan operation and won't send any data out. It will only send out response data for request needing a response after the required operation has been finished. After the request has been processed, RPLIDAR will enter other status specified by the request.

Once RPLIDAR detects something wrong with the device hardware, it will stop the current operation and enter the Protection Stop state. A host system can still communicate with an RPLIDAR in the Protection Stop state to query its working status. But the host system cannot ask the RPLIDAR to perform scan operations unless the host system send a Reset request to reboot the RPLIDAR core system.

Scanning Status

RPLIDAR always checks the motor rotation status when working in the scanning state. Only when the motor rotation speed becomes stable, RPLIDAR will start taking distance measurement and sending out the result data to the host system.
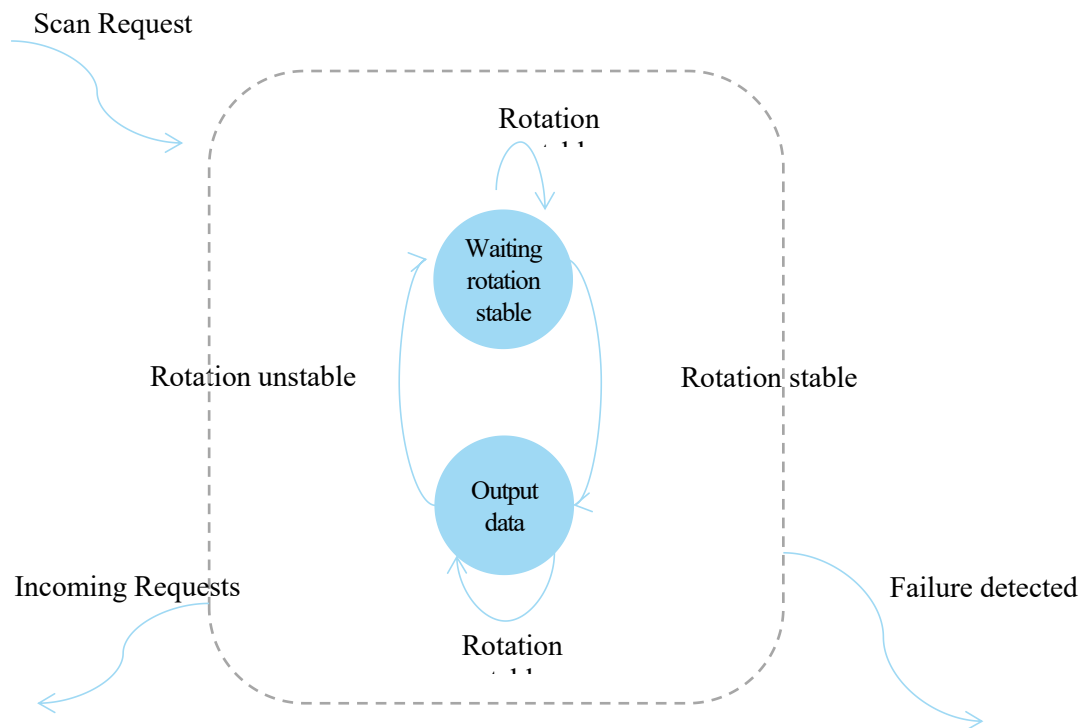

Figure 3-2 RPLIDAR's Working Mode during Scanning

Scan Mode and Measurement Frequency

A new concept called 'scan mode' is introduced since firmware version 1.24. RPLIDAR's following performance may differ in different scan modes:

- Measurement frequency

- Max measurement distance

- Sensitivity of detection

- Environment light elimination

Different model of RPLIDAR support different set of scan modes. Each of them is optimized for specified work environment. Some typical scan modes are defined as below:

Scan Mode Name	Description	Max Sample Rate (sps)	Max Distance	Extra Features
Standard	Traditional Mode	4600 for S1 16000 for S2	16m for S1 16m	
DenseBoost	Dense Mode	9200 for S1 32000 for S2	40m for S1 30m for S2	On the premise of increasing the range limit and sampling rate, the signal quality data output is sacrificed

Figure 3-3 Typical scan modes of RPLIDAR

A new command GET_LIDAR_CONF has been added to help host system to enumerate all scan modes supported by a LIDAR device, as well as the performance parameters of each scan mode. This command can also be used to get "Typical Scan Mode" of a LIDAR, which is the recommend work mode for particular LIDAR model by Slamtec.

To avoid problems, SLAMTEC highly recommends users manipulate work modes of an RPLIDAR via the RPLIDAR Public SDKs.

Requests Overview

All the available requests are listed in the below table. Their detailed descriptions are given in the following sections.

Request Name	Value	Payload	Response Mode	RPLIDAR Operation
STOP	0x25	N/A	No response	Exit the current state and enter the idle state
RESET	0x40	N/A		Reset(reboot) the RPLIDAR core
SCAN	0x20	N/A	Multiple response	Enter the scanning state
EXPRESS_SCAN	0x82	YES		Enter the scanning state and working at the highest speed
GET_INFO	0x50	N/A	Single response	Send out the device info (e.g. serial number)
GET_HEALTH	0x52	N/A		Send out the device health info
GET_SAMPLERATE	0x59	N/A		Send out single sampling time
GET_LIDAR_CONF	0x84	YES		Get LIDAR configuration

Figure 4-1 The Available Requests of RPLIDAR

STOP Request

Request Packet:

A5	25

RPLIDAR will exit the current scanning state once it receives the Stop (0x25) Request sent by a host system. The laser diode and the measurement system will be disabled and the Idle state will be entered. This request will be ignored when RPLIDAR is in the Idle or Protection Stop state.

Since RPLIDAR won't send response packet for this request, host systems should wait for at least 10 millisecond (ms) before sending another request.
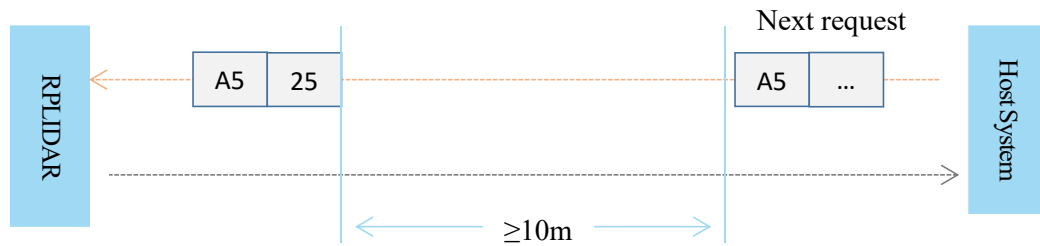
Figure 4-2 The Timing Sequence of a STOP Request

RPLIDAR Core Reset(RESET) Request

Request Packet:

A5	**40**

Host systems can make RPLIDAR core to reset (reboot) itself by sending this request. A reset operation will make RPLIDAR revert to a similar state as it has just been powered up. This request is useful when RPLIDAR has entered the Protection Stop state. After a core reset, RPLIDAR will return to the idle state which will accept the start scan request again.

Since RPLIDAR won't send response packet for this request, host systems should wait for at least 500milliseconds (ms) before sending another request.
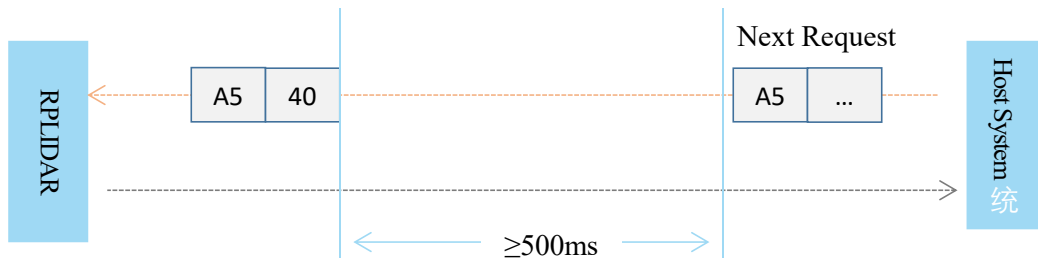

Figure 4-3 The Timing Sequence of a RESET Request

Start Scan（SCAN） Request and Response

Note: RPLIDAR device **models support** 4khz or higher sampling rate will lower the sampling rate when processing this request. Please use EXPRESS_SCAN for the best performance.

This command supports Legacy scan mode only.

Request Packet:	A5	20

Response Descriptor:	A5	5A	05	00	00	40	81

Response Mode: **Multiple**   Data Response Length: **5 bytes**

RPLIDAR, except for the RPLIDAR that is in the Protection Stop State, will enter the scanning state once it receives this request from a host system. Each measurement sample result will be sent out using an individual data response packet. If the RPLIDAR has been in scanning state already, it will stop the current measurement sampling and start a new round of scanning. This request will be ignored when RPLIDAR is in the Protection Stop state.

The related response descriptor will be sent out by RPLIDAR immediately once it receives the request and accepts it. The data response packets related to every measurement sample results will be sent out continuously only after the motor rotation becomes stable. RPLIDAR will leave the Scanning state once it receives a new request from the host system or it detects something is wrong.
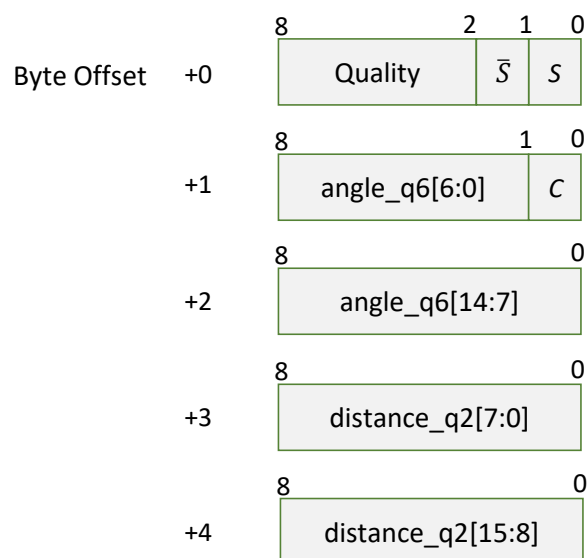
Format of the Data Response Packets:

Byte Offset	+0	8 2 1 0

Byte Offset +0: Quality [8:2], $\bar{S}$ [1], S [0]

+1: angle_q6[6:0] [8:1], C [0]

+2: angle_q6[14:7] [8:0]

+3: distance_q2[7:0] [8:0]

+4: distance_q2[15:8] [8:0]

Figure 4-4 Format of a RPLIDAR Measurement Result Data Response Packet

RPLIDAR encapsulates each measurement sample into a data response packet with the format showed in the above figure and send the packet out. The descriptions of every field within the packet are listed in the following table:

Field Name	Description	Examples / Notes
S	Start flag bit of a new scan	When S is set to 1, the current and incoming packets belong to a new 360° scan.
$\bar{S}$	Inversed start flag bit, always has $\bar{S} = !S$	Can be used as a data check bit.
C	Check bit, constantly set to 1	Can be used as a data check bit.
quality	Quality of the current measurement sample	Related the reflected laser pulse strength.
angle_q6	The measurement heading angle related to RPLIDAR's heading. In degree unit, [0-360) Stored using fix point number.	Refer to the below figure for details. Actual heading = angle_q6/64.0 Degree
distance_q2	Measured object distance related to RPLIDAR's rotation center. In millimeter (mm) unit. Represents using fix point. Set to 0 when the measurement is invalid.	Refer to the below figure for details. Actual Distance = distance_q2/4.0 mm

Figure 4-5 Field Definition of a RPLIDAR Measurement Result Data Response Packet

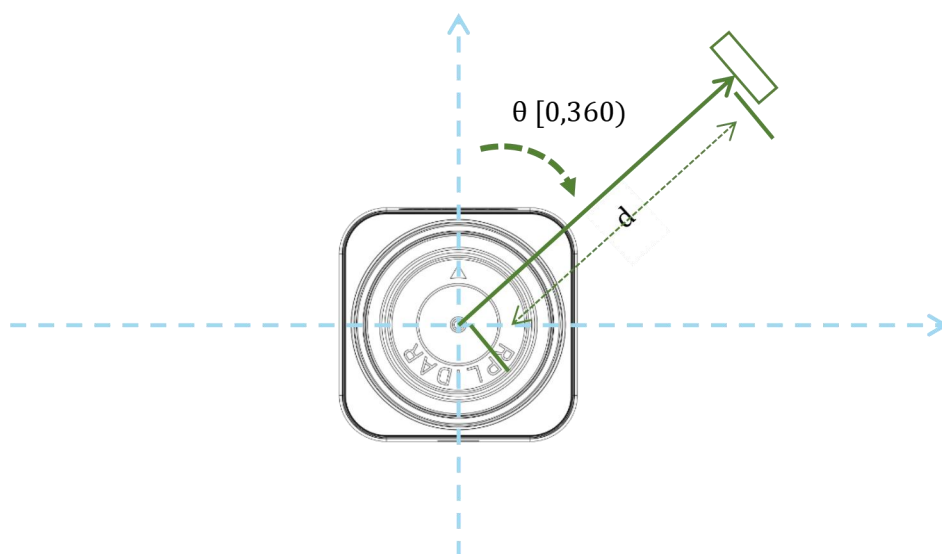The geometric definition of the included angle and distance value is shown as below:

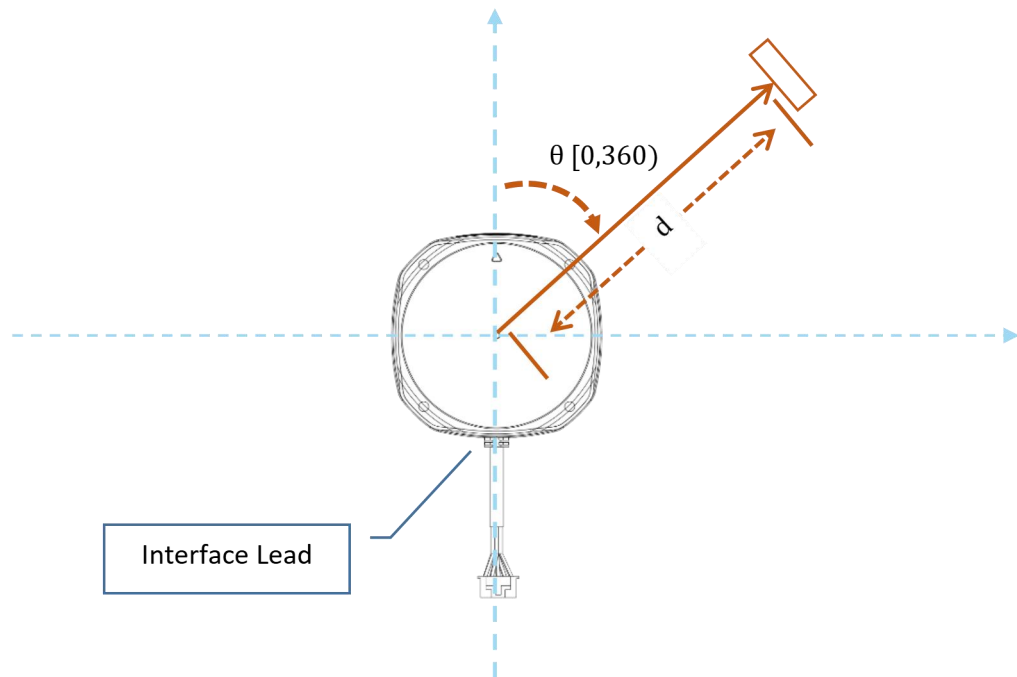*Figure 4-6 Angle and Distance Value Geometric Definition for RPLIDAR S1*

θ [0,360)

d

Interface Lead

Figure 4-7 Angle and Distance Value Geometric Definition for RPLIDAR S2 series
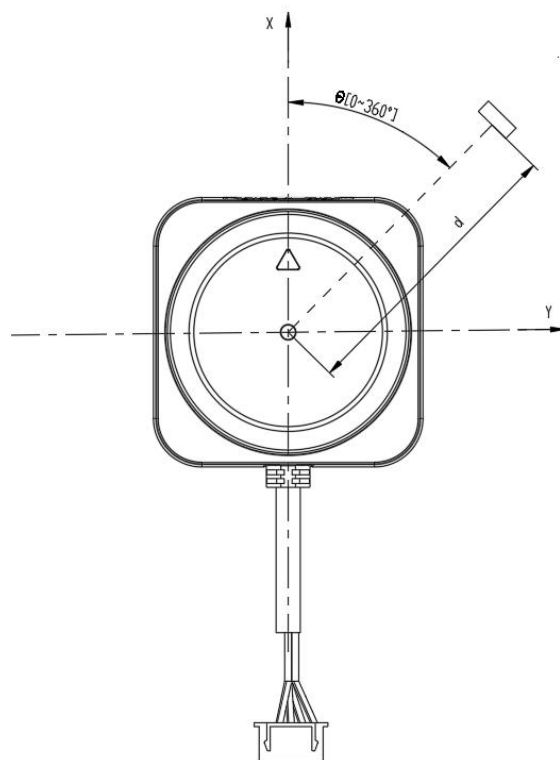

X

θ[0~360°]

d

Y

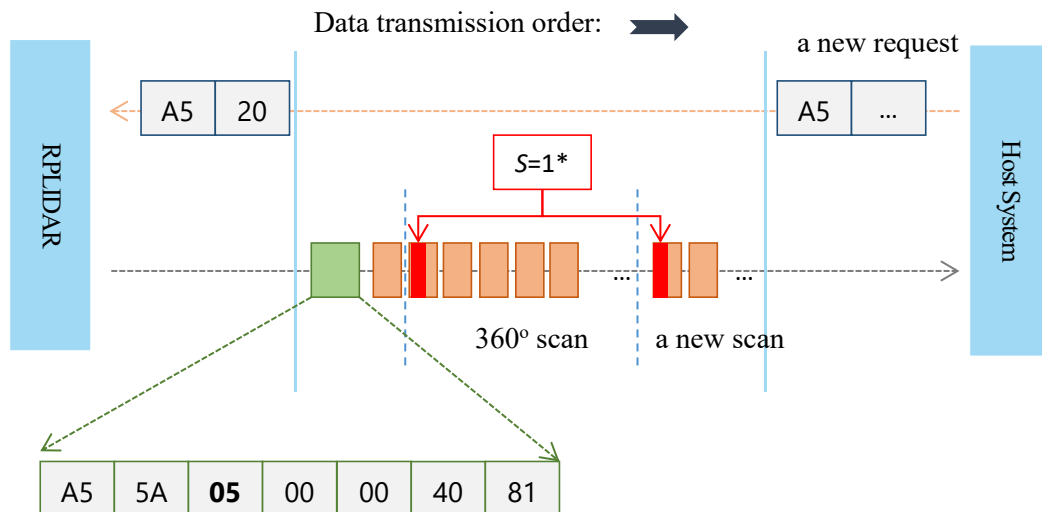*Figure 4-8 Angle and Distance Value Geometric Definition for RPLIDAR S3 series*

Figure 4-9 The Communication Status after Host System Sending a SCAN Request

* S of the first scan point of each scan frame are set to 1, otherwise 0.

Express Scan(EXPRESS_SCAN) Request and Response

Notice: This protocol has been well encapsulated in the supporting open source SDK, and the working mode will be automatically switched according to user requirements. It is recommended to use the relevant APIs of the SDK.

● Dense Version

Request Packet:

A5	82	05	M	00	00	00	00	22

Response Descriptor:

A5	5A	54	00	00	40	85

Response Mode: **Multiple**   Data Response Length: **84 bytes**

RPLIDAR will enter the measurement sampling mode once it receives the express scan(EXPRESS_SCAN) request. Different from the scan(SCAN) request, this request will make RPLIDAR work at the sampling rate as high as it can be. For LIDARs support sampling more than 4000 times per second, the host system should use GET_LIDAR_CONF command to get "Typical Scan Mode", and use this command to make LIDAR work under its best performance and output measurement sample data accordingly.

The host system can use GET_LIDAR_CONF to get all scan modes, as well as performance parameters, such as sample frequency, measurement range and etc. The sample duration of standard and express mode could also be fetched via command GET_SAMPLERATE.

RPLIDAR uses the same state machine and processing logic for this request as the one of the scan(SCAN) request, but uses the different response format.

Format of the request packets:

The express scan(EXPRESS_SCAN) request is required to carry the 5 bytes payload data and the data structure is shown as following. This payload data cannot be omitted.
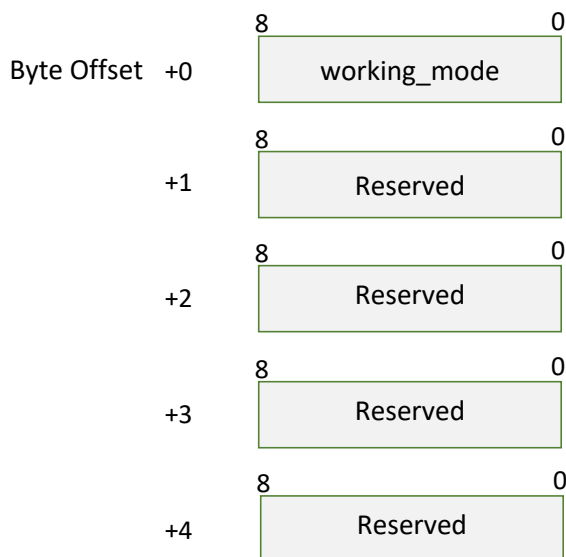

Figure 4-10 Format of a RPLIDAR Express Scan Data Request Packet

The descriptions of every field within the above data are listed in the following table:

Field Name	Description	Examples/Notes
working_mode	The scan mode requested	When set to 0, this command is a legacy version express scan request. When set to value other than 0, this is a extended version express scan request. This value defines the requested work mode which could be fetched via GET_LIDAR_CONF commands.
Reserved	Reserved field, set to 0.	Reserved for future use, set to 0.

Figure 4-11 Field Definition of RPLIDAR Express Scan Data Request Packet

Format of the data response packets (Dense Version):

RPLIDAR may respond in following response packet format according to amount of measurements and the link bandwidth. The actual data packet format could also be fetched via GET_LIDAR_CONF command.
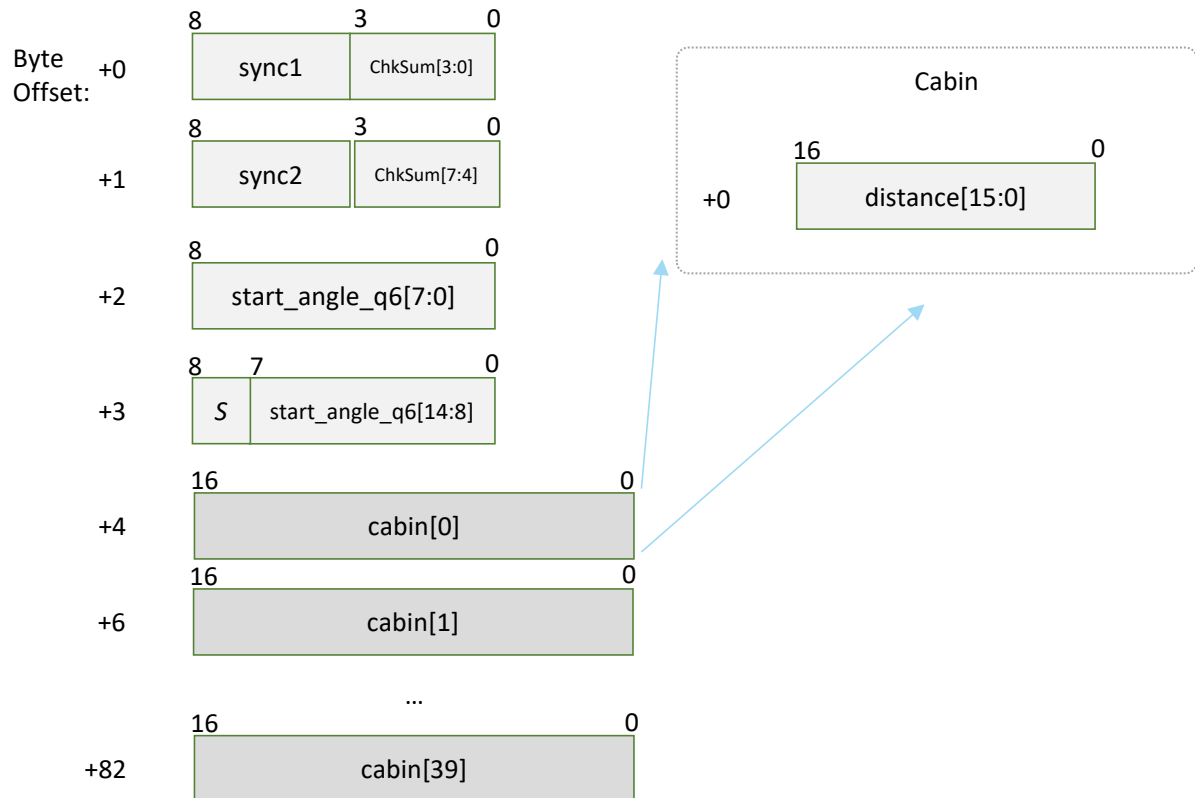

Figure 4-12 Format of a RPLIDAR Express Scan Data Response Packet (Dense Version)

If RPLIDAR had received the request to work in dense scan mode, it would cyclically output the measurement data to host system by the above data packet. The packet contains 40 group of substructures also called Cabin in the same format. Each Cabin is a 2-byte data body corresponding with a measurement sampling data. Therefore, a dense scan data response packet contains 40 measurement sampling data points.

The following table describes the filed definition of the Dense Capsule data.

Field Name	Description	Examples/Notes
sync1	Data packet start sync flag1. Always be 0xA	It is used by host system for identifying the start of a new response packet.
sync2	Data packet start sync flag2. Always be 0x5	It is used by host system for identifying the start of a new response packet.
start_angle_q6	The reference value for the angle data in the current response packet.	Please refer to the following sections for information about the calculated

	Stored with fix point number in q6 format and the unit is degree. Range [0-360)	method of the angle value. It adopts the same coordinate system definition with standard SCAN request. Actual angle =start_angle_q6/64.0 Deg
S	Start flag bit of a new scan	When S is set to 1, the current and incoming packets belong to a new 360° scan.
ChkSum	Obtained by calculating the response data packet using XOR operation to each packet data byte and accumulating them together.	It can be used to determine the validity of a reply packet, starting from start_angle_q6[7:0]
cabin	a 2-byte data body with the angle and distance value of two measurement sampling data sets. A data response packet contains 40 groups of cabin data.	Please refer to the following table for detailed definition.

Figure 4-13 Field Definition of RPLIDAR Express Scan Data Response Dense Capsuled Packet

The filed definition of the Cabin data was defined as the following table.

Field Definition	Description	Examples/Notes
distance	The distance data. When the value is 0, the matched sampling point is invalid.	The unit is millimeter(mm)

Figure 4-14 Field Definition of RPLIDAR Express Scan Dense Cabin Data Response Packet

After receiving the dense express scan request, the communication status between RPLIDAR and host system was illustrated in the following chart.
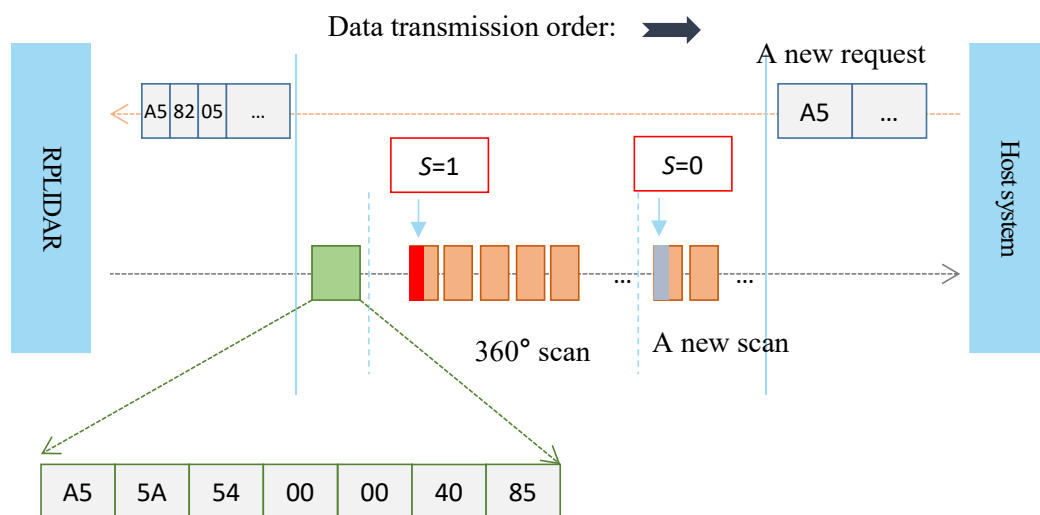

Figure 4-15 The Communication Status Sending Out the Dense Capsuled Express Scan Request

Data Processing of the Express Scan Data (Dense Capsuled Express Scans)

When working in this express scan mode, every sampling distance value is stored in the cabin structural body. The distance value is matched with the actual distance measured in this sampling, and the corresponding angle data can be calculated out with the start_angle_q6 in this request. The detailed calculation method is shown as following:
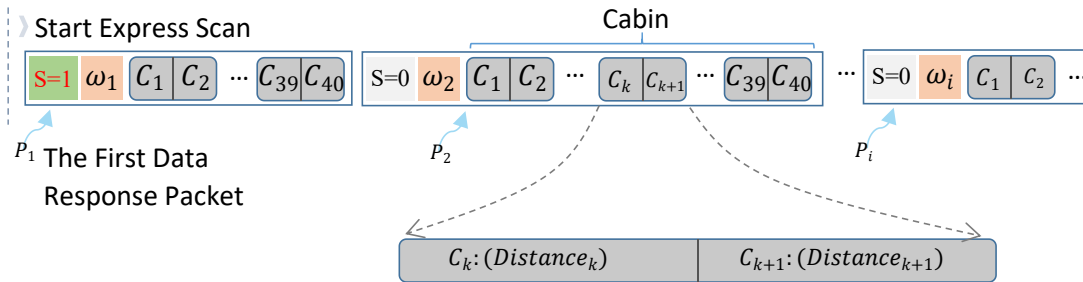

Figure 4-16 The Abstract Description for the Response Data Packet of Express Scan Request

The symbols in the figure are same as preceding section, but measurement sampling point C_k only carries a distance value $Distance_k$. The actual angle θ_k of $Distance_k$ can be obtained as follows:

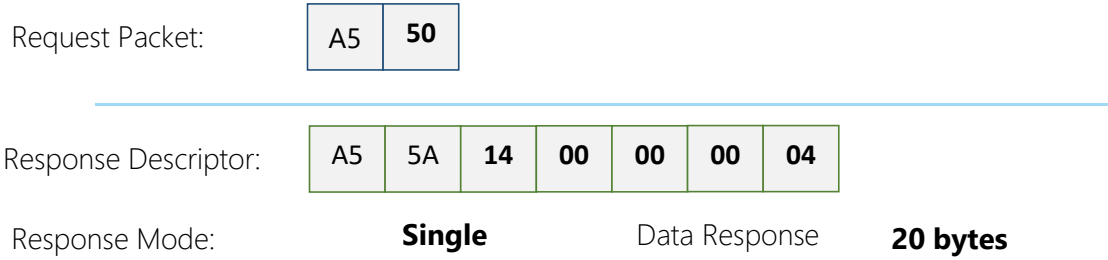$$\theta_k = \omega_i + \frac{AngleDiff(\omega_i, \omega_{i+1})}{40} \cdot k$$

where

$$\text{AngleDiff}(\omega_i, \omega_{i+1}) = \begin{cases} \omega_{i+1} - \omega_i, & \omega_i \leq \omega_{i+1} \\ 360 + \omega_{i+1} - \omega_i, & \omega_i > \omega_{i+1} \end{cases}.$$

Response descriptor flag S:

After entering express scan mode, the first sent data response packet always has the S flag set to 1. In the following measuring process, if the angle value cannot be calculated via the above formula due to rotating speed instability or something wrong, RPLIDAR will reset the flag S. Then the host system is supposed to restart the data analysis based on the current response data packet set as flag S.

Get Device Info (GET_INFO) Request and Response

Request Packet:

A5	50

Response Descriptor:

A5	5A	14	00	00	00	04

Response Mode: **Single**   Data Response   **20 bytes**

RPLIDAR will send out its device information (e.g. serial number, firmware/hardware version) to the host system once it receives this request.
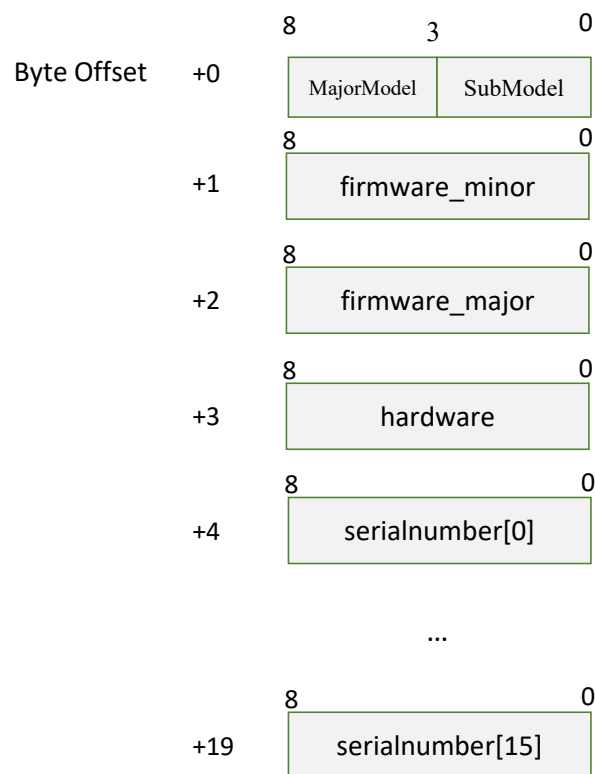
Format of the Device Info Response Packets:

Figure 4-17 Format of a Device Info Data Response Packet

- Each field definition:

Field Name	Description	Examples / Notes
MajorModel	RPLIDAR Major Model	The model ID of the RPLIDAR being used. For S-series lidars, this value needs to be subtracted from the fixed value of 5, such as: 1. The field of S1 corresponds to 6, the field of S2 corresponds to 7
SubModel	RPLIDAR Sub Model	For example, RPLIDAR S2M1 corresponds to a SubModel of 1.
firmware_minor	Firmware version number, the minor value part	The decimal part of the version number
firmware_major	Firmware version number, the major value part	The integer part of the version number
hardware	Hardware version number	
serialnumber[16]	16bit unique serial number	When converting to text in hex, the Least Significant Byte prints first

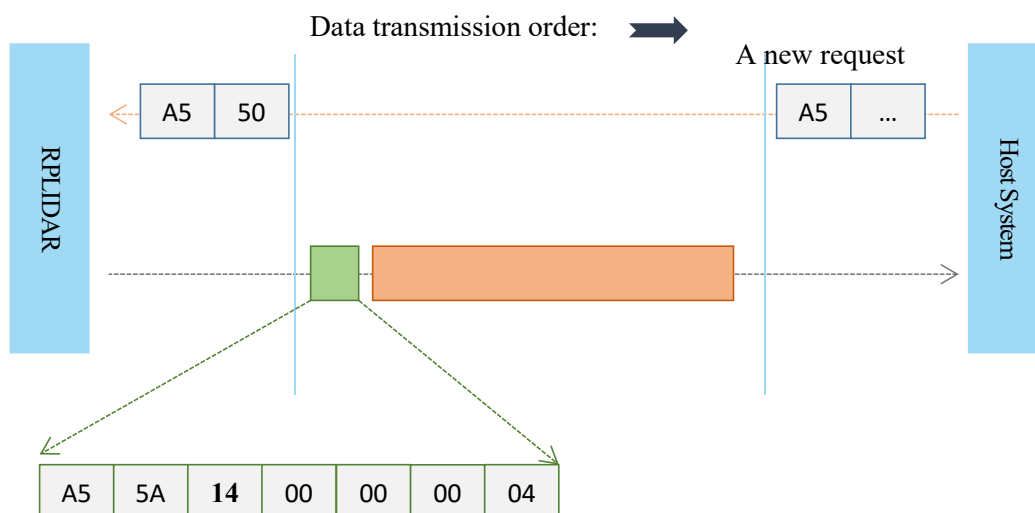*Figure 4-18 Field Definition of Device Info Data Response Packet*

Data transmission order:

Figure 4-19 The Timing Sequence of a GET_INFO Request

Get Device Health Status (GET_HEALTH) Request and Response

Request Packet:

A5	52

Response Descriptor:

A5	5A	3	00	00	00	06

Response Mode: **Single**   Data Response Length: **3 bytes**

A host system can send the GET_HEALTH request to query RPLIDAR's health state. If the RPLIDAR has entered the Protection Stop state caused by hardware failure, the related error code of the failure will be sent out.

Format of the Data Response Packets

Byte Offset:

+0	status
+1	error_code[7:0]
+2	error_code[15:8]
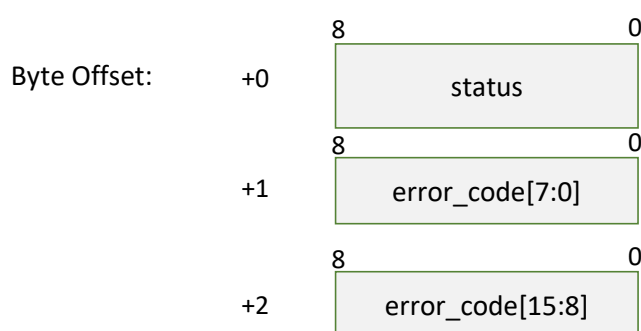
Figure 4-20 Format of a Device Health Data Response Packet

The following table describes the meanings of each field in the preceding packet:

Field Name	Description	Examples / Notes
status	RPLIDAR Health status	Value definition： 0: Good 1: Warning 2: Error When the core system detects some potential risk that may cause hardware failure in the future, the status value will be set to Warning(1). But RPLIDAR can still work as normal. When RPLIDAR is in the Protection Stop state, the status value is set to Error(2).
error_code	Specific warning / Error code	When a warning or error status occurs, the specific error code is recorded in this field.

Figure 4-21 Field Definition of Device Health Status Data Response Packet

When a host system detects RPLIDAR has entered the Protection Stop state, it can set a RESET request to let RPLIDAR core system reboot to escape the Protection

Get sample rate(GET_SAMPLERATE) Request

Request Packet	A5	59

Response Descriptor:	A5	5A	4	00	00	00	15

Response Mode: **Single**   Data Response Length: **4 bytes**
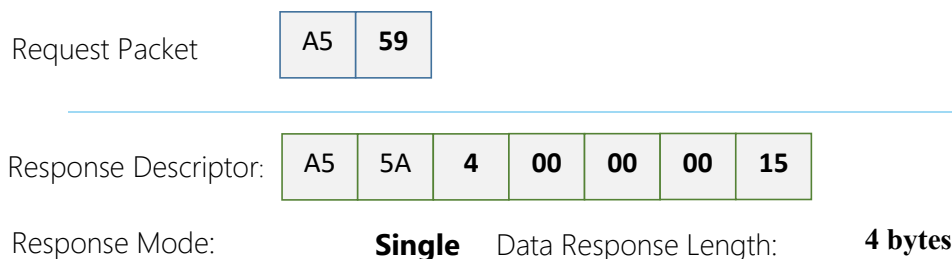
By sending this request, the host system can get the single measurement duration in standard scan mode and express scan mode respectively and calculate the current rotating speed of RPLIDAR accurately.
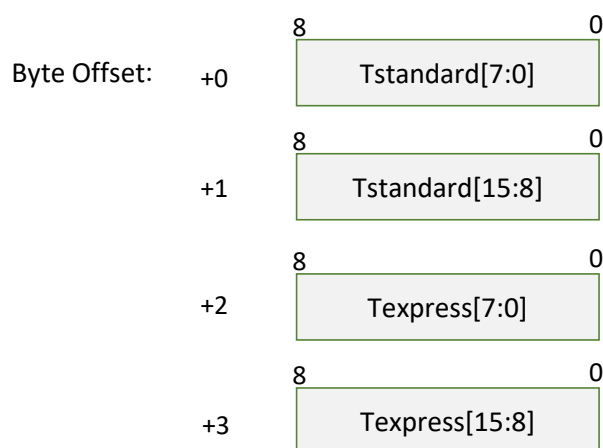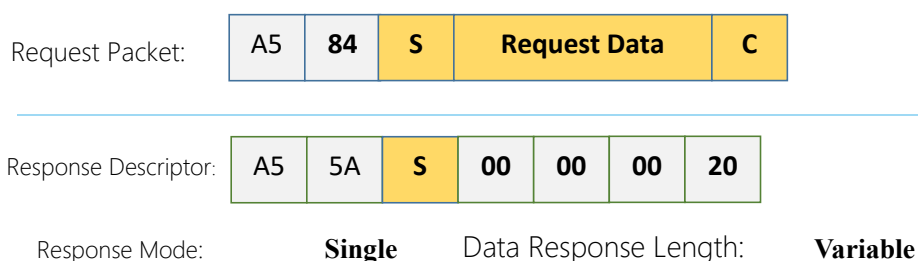
Format of the Data Response Packets

Byte Offset:

	8 ... 0
+0	Tstandard[7:0]
+1	Tstandard[15:8]
+2	Texpress[7:0]
+3	Texpress[15:8]

Figure 4-22 Format of a Sample Rate Data Response Packet

The following table describes the filed definition of the above packet.

Field Name	Description	Examples/Notes
Tstandard	In scan(SCAN) mode, the time used when RPLIDAR takes a single laser ranging. Unit: microsecond(uS)	It can be used for debugging the rotating speed when the RPLIDAR uses the SCAN request.
Texpress	In express scan(EXPRESS_SCAN) mode, the time used when RPLIDAR takes a single laser ranging Unit: microsecond(uS)	It can be used for debugging the rotating speed when the RPLIDAR uses the EXPRESS_SCAN request.

Figure 4-23 Field Definition of Sample Rate Data Response Packet

Device configuration query command (GET_LIDAR_CONF)

Request Packet:

A5	84	S	Request Data	C

Response Descriptor:

A5	5A	S	00	00	00	20

Response Mode: **Single**   Data Response Length: **Variable**

Use this command to fetch characters of RPLIDAR. Use type and payload to specify the configuration entry to query.

Format of data request packets:
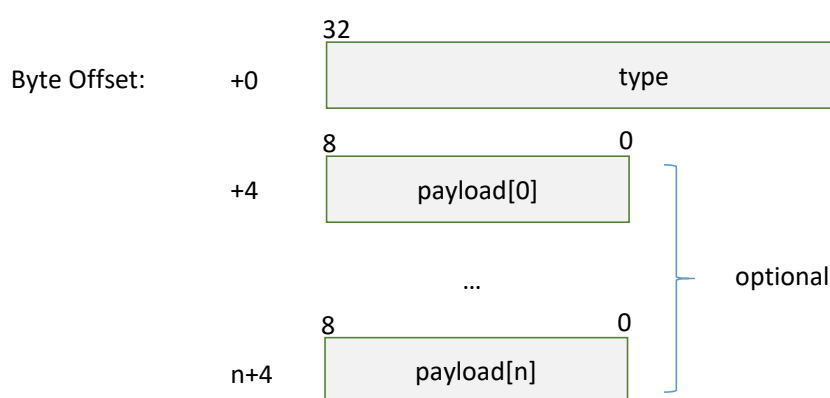
Figure 4-24 Format of a Get LIDAR Conf Request Packet

Fields of above packet format are defined below:

Field Name	Description	Samples/Notes
type	The configuration entry to query	Please refer to Figure 4-30 for predefined configurations
Payload[n]	Additional parameters to above entry	Optional. Defined by each specific configuration type.

Figure 4-25 Definition of Get LIDAR Conf Request Fields

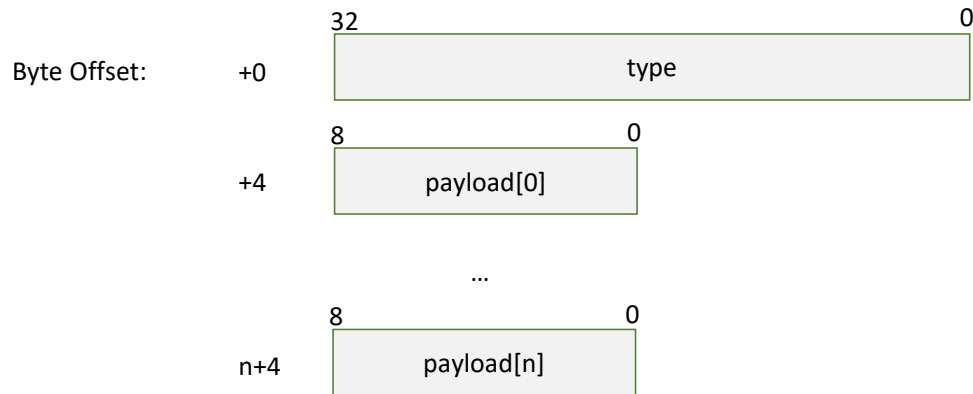## Format of data response packets:

Figure 4-26 Format of a Get LIDAR Conf Response Packet

Fields of above packet format are defined below:

Field Name	Description	Samples/Notes
type	The configuration entry id	Same as 'type' field in the request field.
Payload[n]	Configuration value (varies between different configuration entries)	Please refer to definition of each configuration entry for detailed format and information.

Figure 4-27 Definition of Device Configuration Get Command Response Packet

Fundamental Data Type Definitions

This command uses following fundamental data types or their combination as payload. All data are stored in little-endian byte order.

Type name	Description	Length (bytes)
u8, u16, u32, u64	Unsigned integer whose word length is indicated by the numeric suffix of its type name (number of bits)	1, 2, 4, 8
s8, s16, s32, s64	Signed integer whose word length is indicated by the numeric suffix of its type name (number of bits)	1, 2, 4, 8
string	String in UTF-8 encoding (end with 0, no BOM header)	Variable
float	Single precision float point value in IEEE-754 format	4
double	Double precision float point value in IEEE-754 format	8

Figure 4-28 Fundamental Data Types and Definition

Available configuration entries

Type	Description	Request Payload	Response Payload
0x70	RPLIDAR_CONF_SCAN_MODE_COUNT Get the amount of scan modes supported by the LIDAR	None	u16
0x71	RPLIDAR_CONF_SCAN_MODE_US_PER_SAMPLE Get microsecond cost per measurement sample for specific scan mode(in Q8 fixed point format)	u16	u32
0x74	RPLIDAR_CONF_SCAN_MODE_MAX_DISTANCE Get max measurement distance for specific scan mode (in m, Q8 fixed point format)	u16	u32
0x75	RPLIDAR_CONF_SCAN_MODE_ANS_TYPE Get the answer command type for this scan mode	u16	u8
0x7C	RPLIDAR_CONF_SCAN_MODE_TYPICAL Get the typical scan mode id of LIDAR	None	u16
0x7F	RPLIDAR_CONF_SCAN_MODE_NAME Get a user friendly name for the scan mode	u16	string

Figure 4-29 Supported Configuration Entries

RPLIDAR_CONF_SCAN_MODE_COUNT (0x70) Configuration Entry

RPLIDAR returns the amount of scan modes supported when receives this command. RPLIDAR supports scan mode ids from 0 to (scan_mode_count – 1).

For instance, device returning 2 according to this query means that the device support 2 work modes, whose ids are 0, 1. The host system may use the work mode id and other configuration type to get specific characters of the work mode.

RPLIDAR_CONF_SCAN_MODE_US_PER_SAMPLE (0x71) Configuration Entry

Get sample duration of the scan mode, whose id is specified by the payload of request.

When the request packet is valid and the specified scanning mode ID is supported by the current device, RPLIDAR will return the laser range time of the device in that mode. The physical unit of its value is microsecond (uS).

RPLIDAR_CONF_SCAN_MODE_MAX_DISTANCE (0x74) Configuration Entry

Get max measurement distance of the scan mode, whose id is specified by the payload of request.

When the request packet is valid and the specified scan mode ID is supported by the current device, RPLIDAR returns the maximum measurement radius distance supported by the device in that mode. The physical unit of the value is the meter and is stored in q8 fixed-point decimal mode.

RPLIDAR_CONF_SCAN_MODE_ANS_TYPE (0x75) Configuration Entry

Get the answer command type of the scan mode, whose id is specified by the payload of request.

The return value is 8bit unsigned int, denotes the answer command type.

Typical return answer types:

- 0x81 –In this mode, the scan range data is returned using SCAN command corresponding to the response data format

- 0x82 – In this mode, the scan range data is returned using EXPRESS_SCAN command corresponding to the traditional version format

- 0x83 – – In this mode, the scan range data is returned using EXPRESS_SCAN command corresponding to the ultra capsuled format

RPLIDAR_CONF_SCAN_MODE_TYPICAL (0x7C) Configuration Entry

This request returns an ID value for the best scanning mode recommended by the current RPLIDAR device. It is recommended to use the operating mode ID returned by this field to drive lidar data acquisition to obtain reliable performance unless the characteristics of the operating mode supported by the current RPLIDAR device are specifically understood.

RPLIDAR_CONF_SCAN_MODE_NAME (0x7F) Configuration Entry

This request returns an ID value for the best scanning mode recommended by the current RPLIDAR device. It is recommended to use the operating mode ID returned by this field to drive lidar data acquisition to obtain reliable performance unless the characteristics of the operating mode supported by the current RPLIDAR device are specifically understood.
When the request packet is valid and the specified scanning mode ID is supported by the current device, RPLIDAR will return the text representation of the current mode corresponding to the requested mode ID that is convenient for human reading and understanding.

Device motor speed control command (MOTOR_SPEED_CTRL)4

Request Packet:

A5	A8	02	Rpm	C

A host system can send this command to control the real-time motor speed of RPLIDAR. And RPLIDAR will be enter the idle state only if have received the command with Rpm = 0.

Format of the Data Request Packets:

Byte Offset: +0

```
16                                                        0
+--------------------------------------------------------+
|                         Rpm                            |
+--------------------------------------------------------+
```
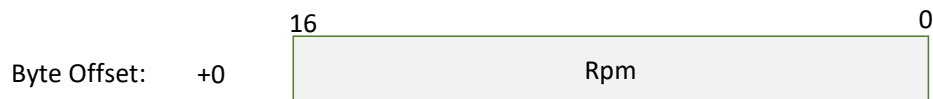
Figure 4-30 Format of a MOTOR_SPEED_CTRL Request Packet

Typical work flow of retrieving scanning data from an RPLIDAR

It is recommended that a host system always follows the below sequence to enable RPLIDA's scanning operation and retrieve the scanning data. Before sending a SCAN request, the host system should send a GET_HEALTH request in advance to query the RPLIDAR's health status. In case RPLIDAR is in the Protection Stop state, the host system can send a RESET request to try to escape the Protection Stop state. Please refer to the SDK code for implementation details.
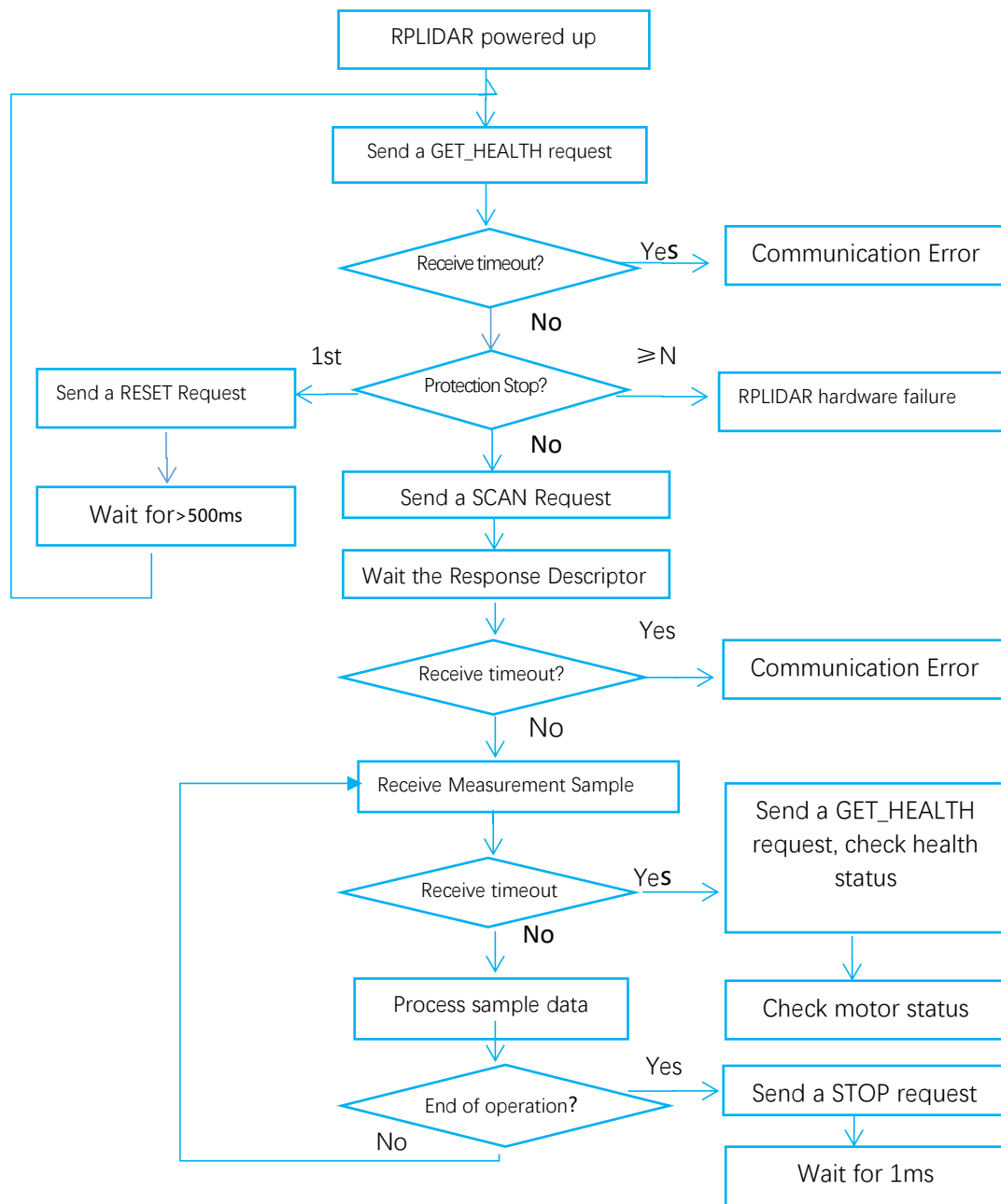
Figure 5-1 Recommendation for Starting RPLIDAR Scanning and Data Retrieving

Calculate RPLIDAR Scanning Speed

In most cases, external systems do not need to care about the actual scanning speed of RPLIDAR. RPLIDAR S series have internal scanning speed detection and closed-loop control system, which can stabilize the current rotation speed in real time and ensure the accuracy of ranging results.

The external system can record the receiving time interval ΔT of two data response messages whose initial flag bit is 1(S=1) after RPLIDAR starts ranging sampling. It represents the time it takes for the current RPLIDAR ranging core to rotate once. The actual speed can be obtained by the following formula:

$$\text{RPM} = \frac{1}{\Delta T} * 60$$

The calculated value can be used as a feedback to control the motor speed.

Date	Description
2013-3-5	Initial version
2014-1-25	Modified related descriptions
2014-3-8	Added descriptions about the time requirement of sending request packet.
2015-8-21	Modified the incoherence in the context of GET_HEALTH
2016-4-10	Added descriptions for the RPLIDAR A2 newly added protocol
2016-5-4	Fixed a description bug
2016-10-28	Fixed a description bug in the EXPRESS_SCAN protocol section
2017-05-15	Release 1.0 Version
2018-03-08	Added notes for higher class scan modes in Express Mode protocol
2018-11-11	Added notes for GET_LIDAR_CONF and Ultra Capsuled Mode protocol
2019-03-28	Added note for Dense Capsuled Mode protocol
2021-02-25	Add S1 scan mode to Figure 3-3
2021-11-25	Add S2 scan mode to Figure 3-3
2022-06-25	Modified the Request Packet of Dense Version
2023-06-12	Some contents were corrected, and the corresponding contents of RPLIDAR A series were deleted
2023-07-24	Add RPLIDAR S3 model
2023-07-24	Add Figure 4-8 Angle and Distance Value Geometric Definition for RPLIDAR S3 series

Image and Table Index