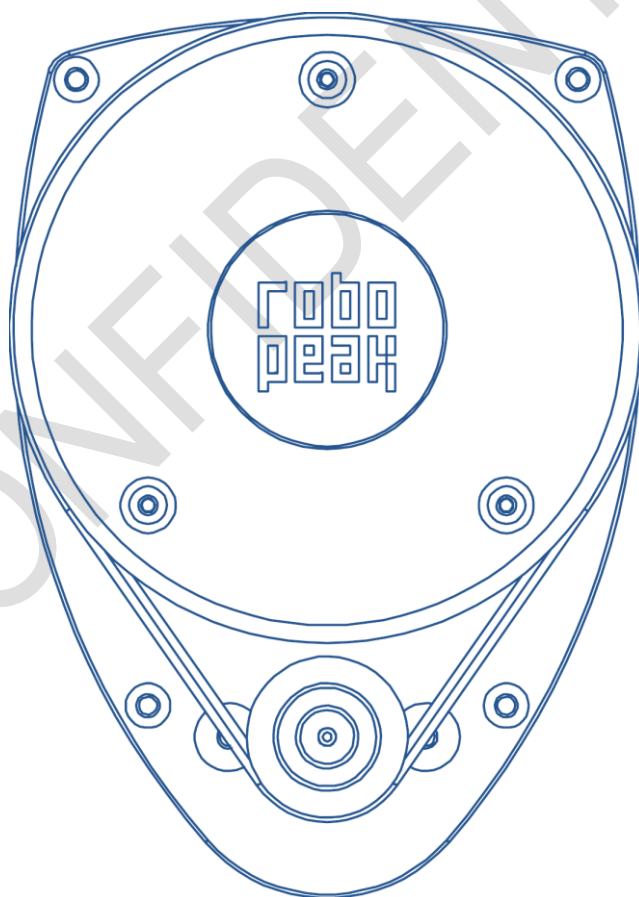




## RPLIDAR

### Low Cost 360 degree 2D Laser Scanner (LIDAR) System Interface Protocol and Application Notes

2014-3



## Contents:

---

1. Overview .....	3
SDK and Demo Program .....	3
2. Protocol Basics .....	4
Basic Communication Mode .....	4
Request/Response Modes .....	4
Request Packets' Format.....	7
Response Packets' Format .....	8
3. State Machine of RPLIDAR's working mode .....	10
Major Working States and Translation Conditions .....	10
Scanning States .....	11
4. Requests and Response Data .....	12
Requests Overview .....	12
STOP Request.....	12
RPLIDAR Core Reset (RESET) Request.....	13
Start Scan (SCAN) Request and Response .....	14
Force Scan (FORCE_SCAN) Request and Response .....	17
Get Device Info (GET_INFO) Request and Response .....	17
Get Device Health Status (GET_HEALTH) Request and Response.....	19
5. Application Notes .....	21
Retrieving scanning data from an RPLIDAR.....	21
Calculate RPLIDAR Scanning Speed .....	22



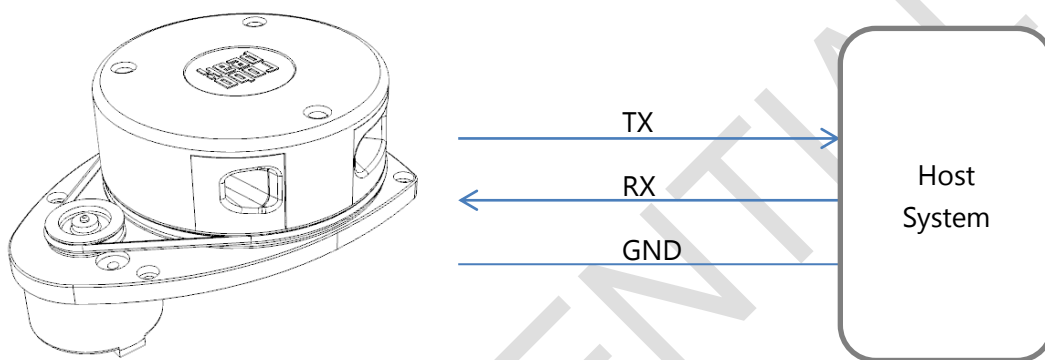
6. Revision History.....	24
--------------------------	----

CONFIDENTIAL

## 1. Overview

RPLIDAR uses a packet based protocol to transmit data between a host system and the core. In order to retrieve the scan results and control RPLIDAR, host systems are required to communicate with RPLIDAR via the TTL UART interface provided by the RPLIDAR core system.

In this document, a detailed interface protocol specification used in RPLIDAR is provided to help developers understand how to retrieve scan data, device status and health information etc. and how to manipulate RPLIDAR's working mode.



The low level UART interface specifications such as baud-rate and logical signal level are described in the RPLIDAR datasheet. Please make sure you have read the datasheet in advance.

### SDK and Demo Program

RoboPeak provides open-sourced SDK and demo program to help customers integrate RPLIDAR into their systems quickly. The SDK implements all the communication stacks, driver logics and related data structures described in this document.

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

Please refer to the SDK manual for details.

## 2. Protocol Basics

---

### Basic Communication Mode

RPLIDAR uses a packet based protocol to communicate with host systems. All the packets transmitted on the interface channel are binary data based and share uniformed 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 perform related operations required by a host system once it received 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 it received the request and scan result data is sent out to the host system continuously.

### Request/Response Modes

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

#### 1) Single request - Single response

This mode is commonly used when a host system is asking an RPLIDAR for data. Once an RPLIDAR receives such type of requests using this mode, it should reply the host system with a single response packet after some necessary operation has been done.

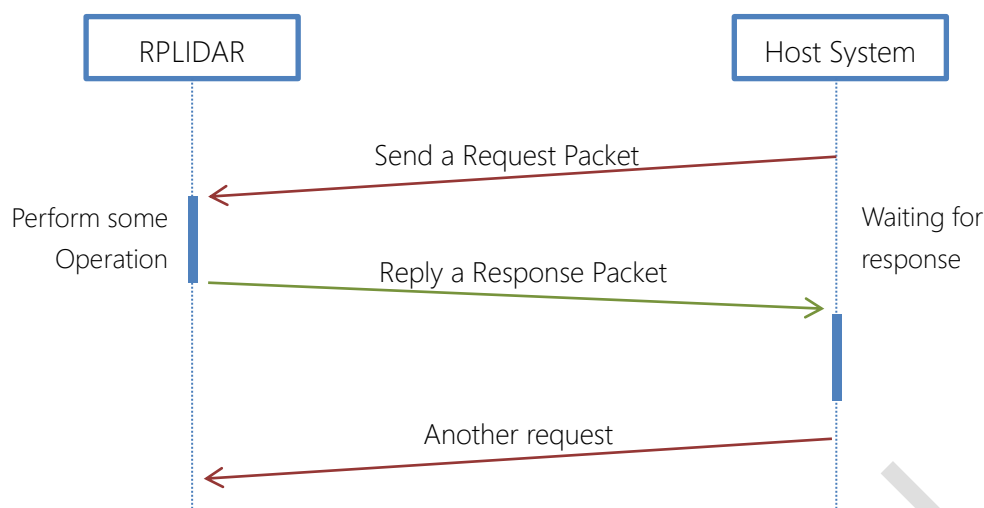


Figure 1 Single Request - Single Response Mode

Host system should prevent sending extra request packets if the RPLIDAR is busy with handing the current request and hasn't reply 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.

## 2) Single request – Multiple response

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.

This mode is used when asking RPLIAR 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.

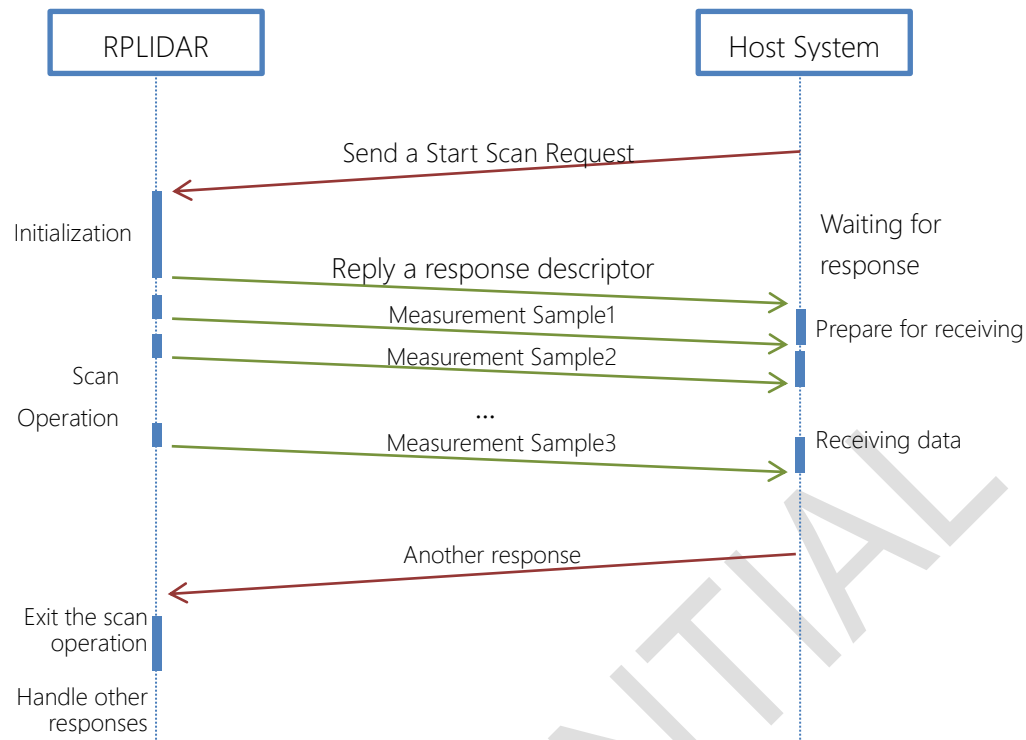


Figure 2 the Single Request - Multiple Response mode used in RPLIDAR's scan operation

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

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.

### 3) Single Request – No Response

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

Host systems should wait for some 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.

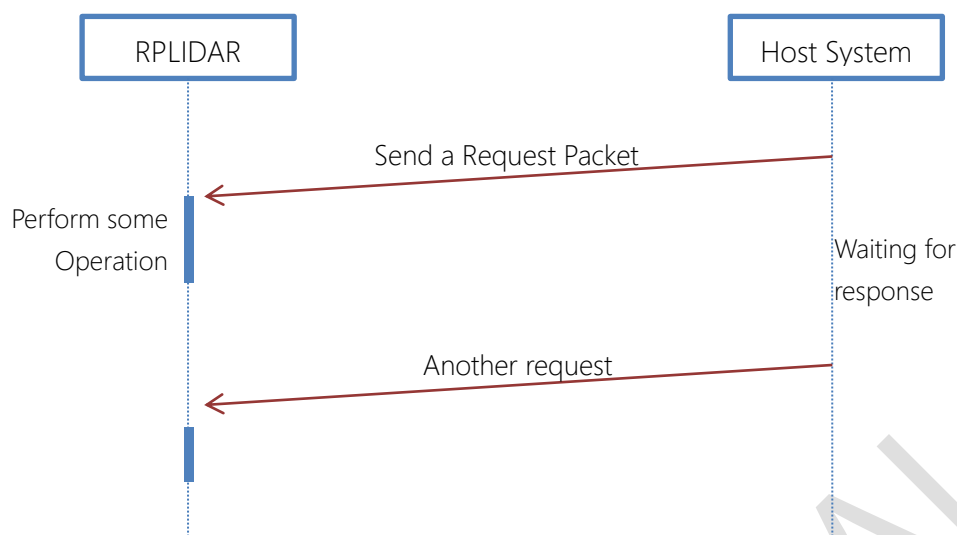
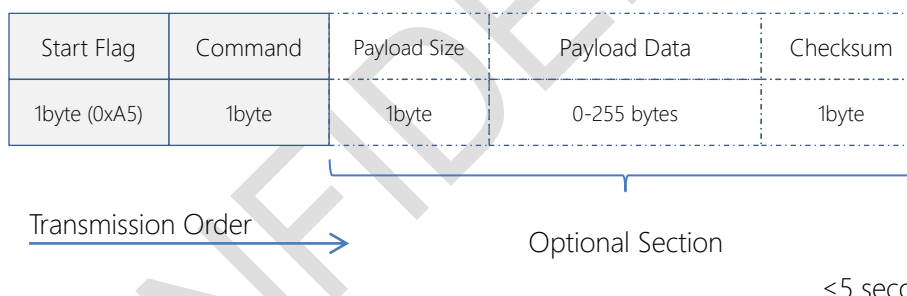


Figure 3 Single Request - No Response Mode

### Request Packets' Format

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



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:

$$\text{checksum} = 0 \oplus 0xA5 \oplus \text{CmdType} \oplus \text{PayloadSize} \oplus \text{Payload}[0] \oplus \dots \oplus \text{Payload}[n]$$

- Timing Consideration

All bytes within a request packet must to 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 required 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 information of the incoming data responses. All the response descriptors share a same format.

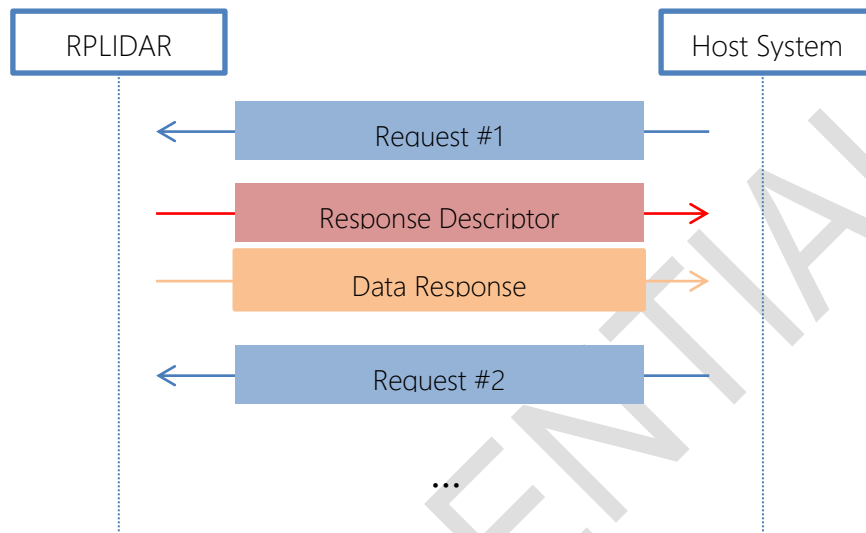


Figure 4 Response packets sent during a Single Request - Single Response mode

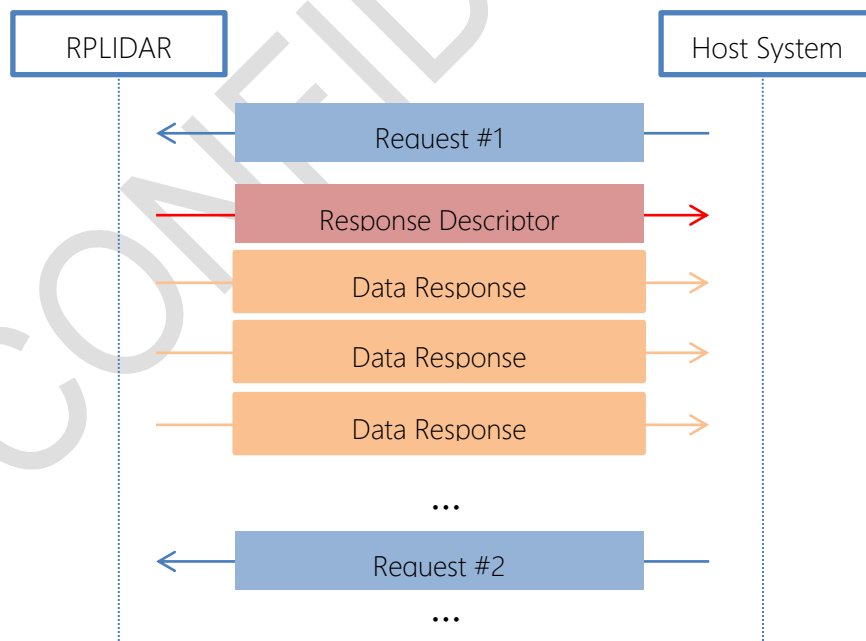


Figure 5 Response packets sent during a Single Request - Multiple Response mode

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 →

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 continuous send out data response packets with the same format in the current session.
0x2	Reserved for future use
0x3	Reserved for future use

The 1byte Data Type field 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.

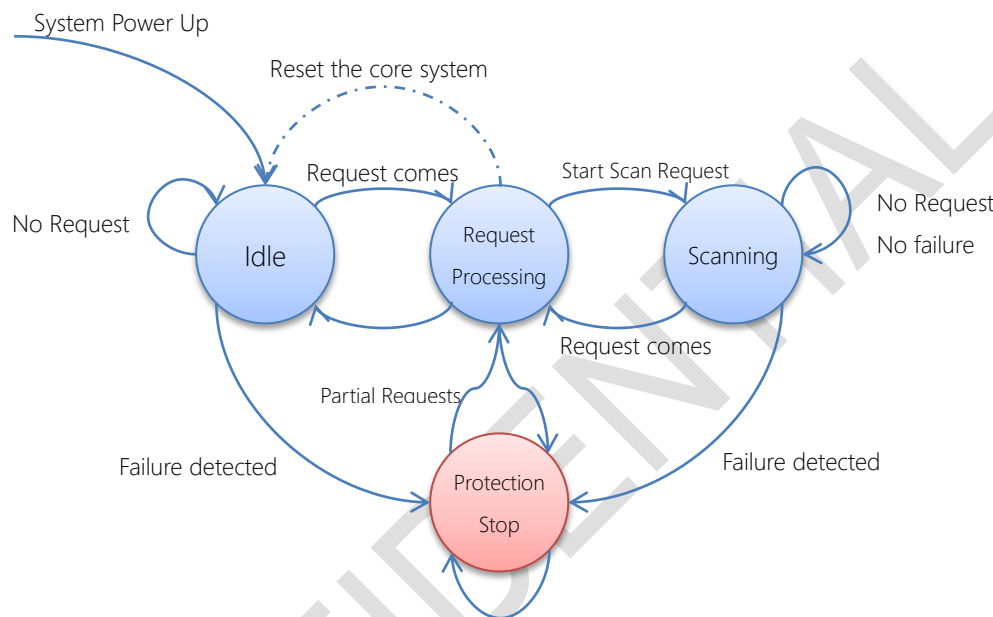
Different from response descriptors, there is no common format used among response data packets. Each type of response data has its own data format and packet length based on its type.

### 3. State Machine of RPLIDAR's working mode

#### Major Working States and Translation 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:



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, 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 when the required operation has been finished and the request asks RPLIDAR to reply to the host system. 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 States

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

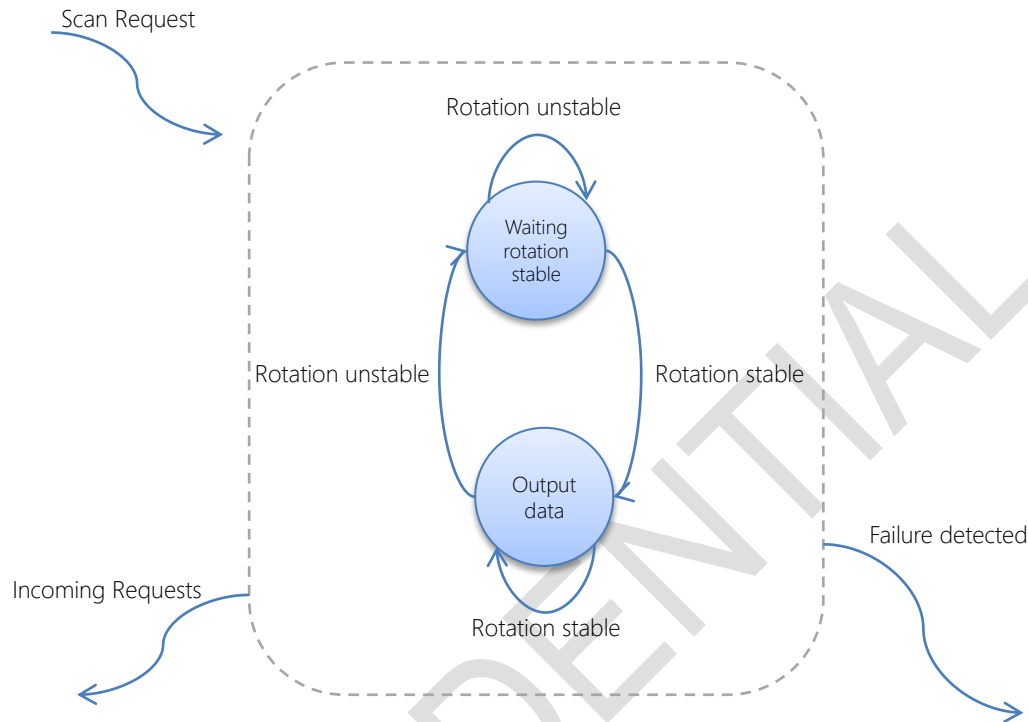


Figure 6 Sub-state translations of the Scanning State

## 4. Requests and Response Data

### 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
FORCE_SCAN	0x21	N/A		Enter the scanning state and force data output without checking rotation speed
GET_INFO	0x50	N/A	Single response	Send out the device info (e.g. serial number)
GET_HEALTH	0x51	N/A		Send out the device health info

### 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 Protection Stop state.

Since RPLIDAR won't send response packet for this request, host systems should wait for at least 1 millisecond (ms) before sending another request.

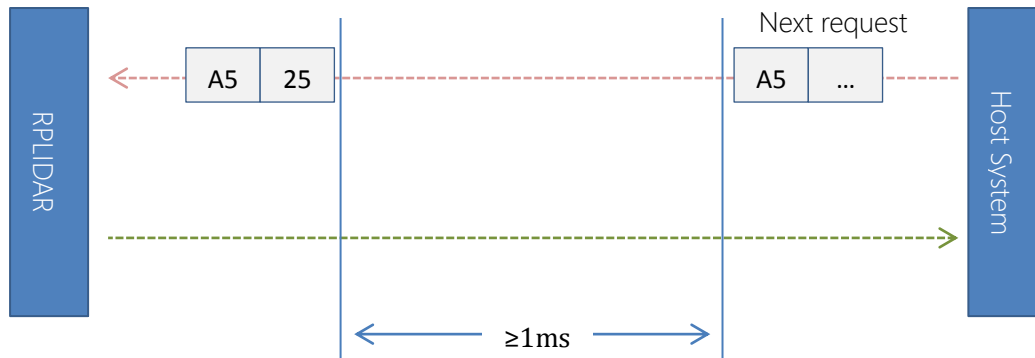


Figure 7 the timing sequence of a STOP request

## RPLIDAR Core Reset (RESET) Request

Request Packet:

A5	<b>40</b>
----	-----------

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 2 milliseconds (ms) before sending another request.

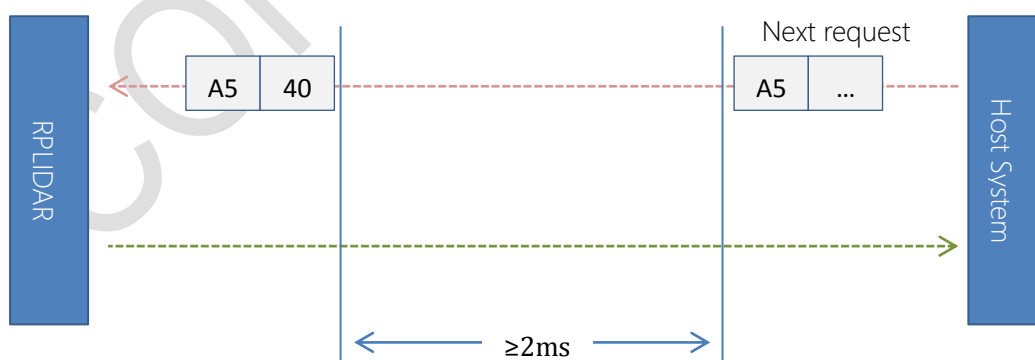


Figure 8 the timing sequence of a RESET request

## Start Scan (SCAN) Request and Response

Request Packet:

A5	20
----	----

Response Descriptor:

A5	5A	05	00	00	40	81
----	----	----	----	----	----	----

Response Mode:

**Multiple**

Data Response Length:

**5 bytes**

RPLIDAR will enter the scanning state once it receives this request from a host system except for the Protection Stop State. Each measurement sample result will be sent out using an individual data response packet. 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:

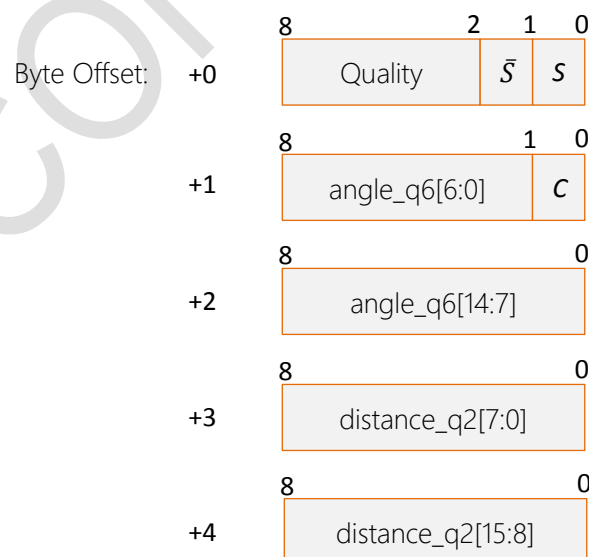


Figure 9 Format of a measurement result data response packet

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 = $\text{angle\_q6}/64.0$ 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 = $\text{distance\_q2}/4.0$ mm



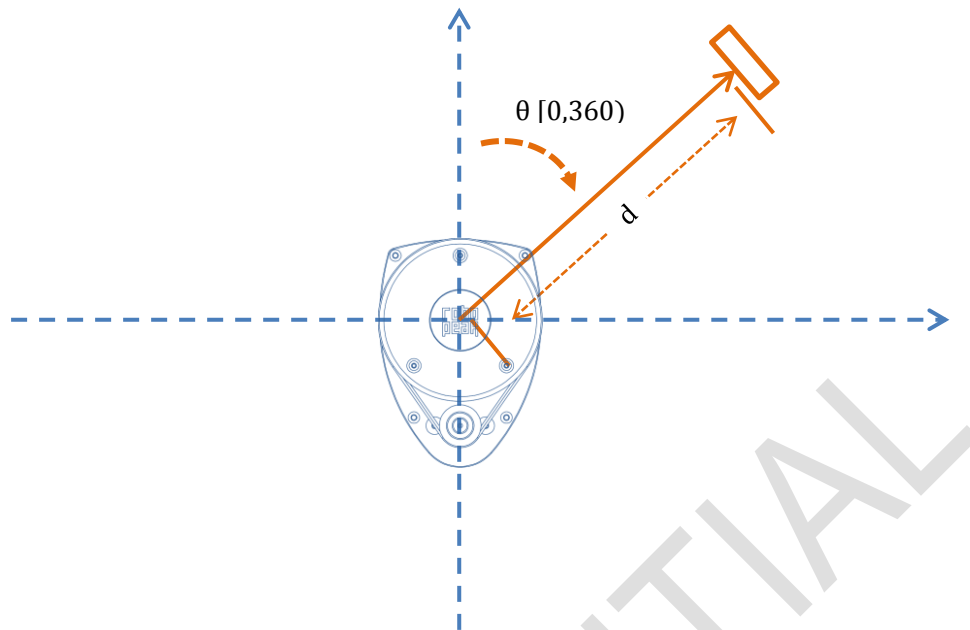


Figure 10 the geometry definition of the heading angle and object distance value

The communication timing sequence of a SCAN request and its related responses:

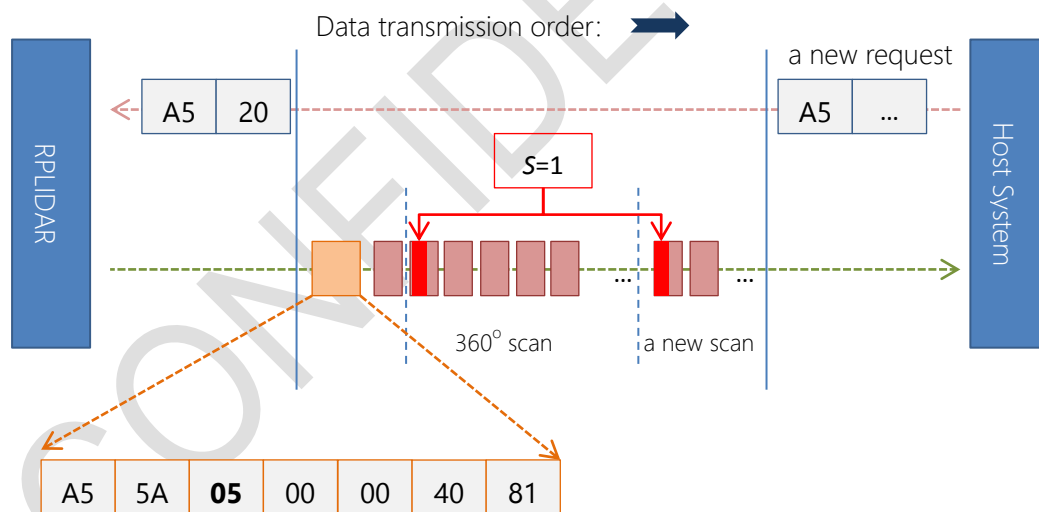


Figure 11 the timing sequence of a SCAN request of its responses

## Force Scan (FORCE\_SCAN) Request and Response

Request Packet:

A5	21
----	----

Response Descriptor:

A5	5A	05	00	00	40	81
----	----	----	----	----	----	----

Response Mode:

**Multiple**

Data Response Length:

**5 bytes**

A force scan (FORCE\_SCAN) request forces RPLIDAR to start measurement sampling and send out the results immediately once it receives this request. The motor speed checking logic will be bypassed. This request is useful for device debugging.

RPLIDAR use the same response format as the one of the SCAN request.

## 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 Length:

**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:

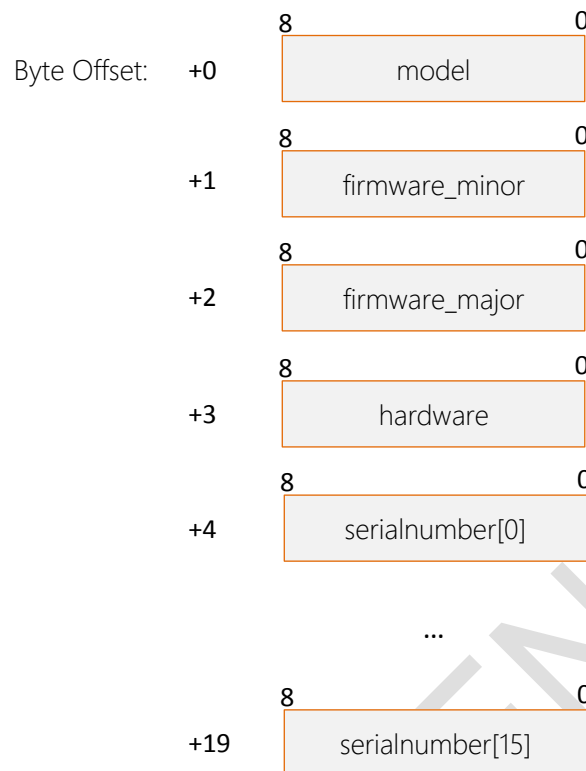


Figure 12 Format of a device info data response packet

Field Name	Description	Examples / Notes
model	RPLIDAR model ID	The model ID of the RPLIDAR being used
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]	128bit unique serial number	When converting to text in hex, the Least Significant Byte prints first

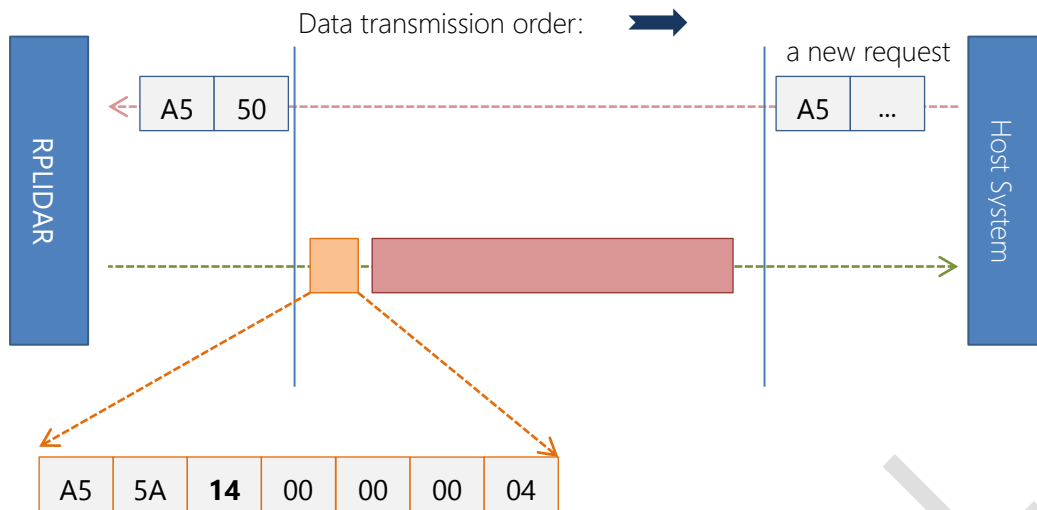


Figure 13 the timing sequence of a GET\_INFO request and its response

## 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:

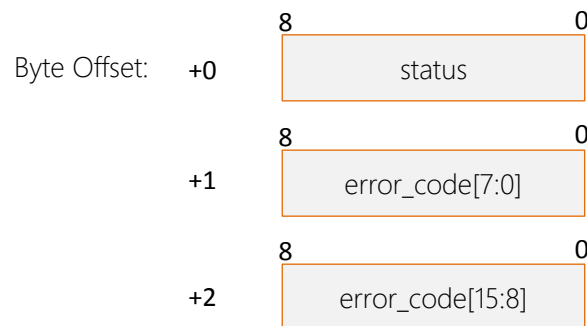


Figure 14 Format of a device health data response packet

Field Name	Description	Examples / Notes
status	RPLIDAR Health State	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	The related errorcode that caused a warning/error.	

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 Stop state. However, if RPLIDAR enters the Protection Stop state for several times, this may be a sign of some unrecoverable damage has occurred in RPLIDAR.

## 5. Application Notes

---

### Retrieving scanning data from an RPLIDAR

It is recommended that a host system always follows the below sequence to enable RPLIDAR'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.

CONFIDENTIAL

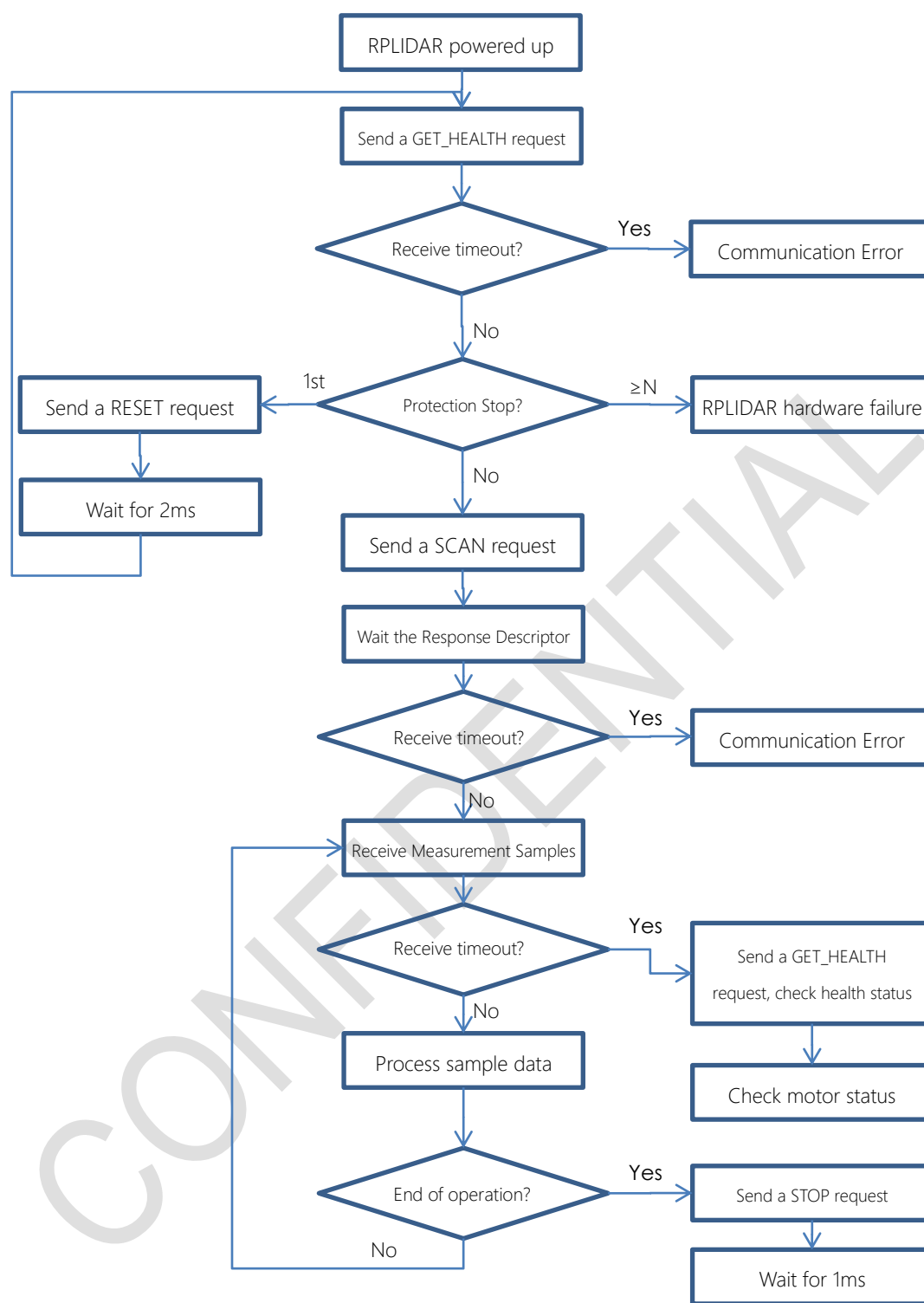


Figure 15 the recommended RPLIDAR operating sequence

Please refer to the SDK code for implementation details.

## Calculate RPLIDAR Scanning Speed

In most cases, there is no need for a host system to care about the actual RPLIDAR scanning speed. RPLIDAR ensures the precision of the measurement results using its

build-in self-adaptive motor rotation speed detector.

If an RPLIDAR is required to have a fixed scanning speed, the host system can control the motor speed using a PWM driver and based on the speed feedback provided by the algorithm described in this section.

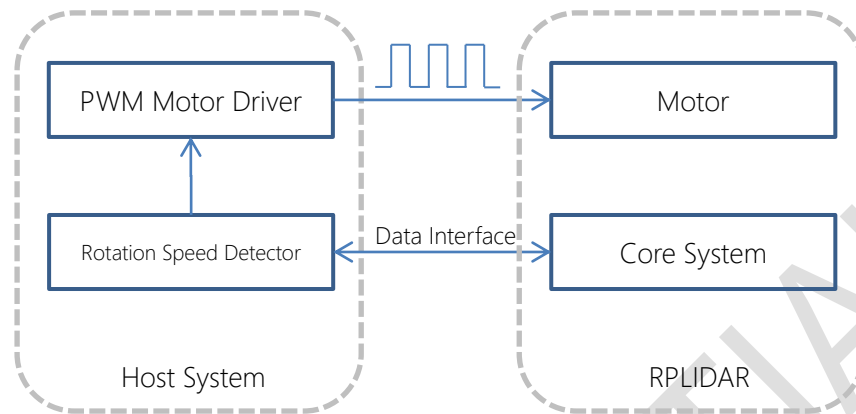


Figure 16 block diagram of RPLIDAR speed detection and control system

The host system can keep recording the interval time between two adjacent measurement sample data response with the start flag bit S set to 1 (S=1), called  $\Delta T$ . The interval time represents how long the RPLIDAR has spent to perform a 360° scan. So the actual scan speed can be calculated using the following equation:

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

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



## 6. Revision History

---

Date	Content
2013-3-5	Initial draft
2014-1-25	1. Translated into English 2. Refined some description
2014-3-8	Added description to the timing requirements of transmitting request packets

CONFIDENTIAL