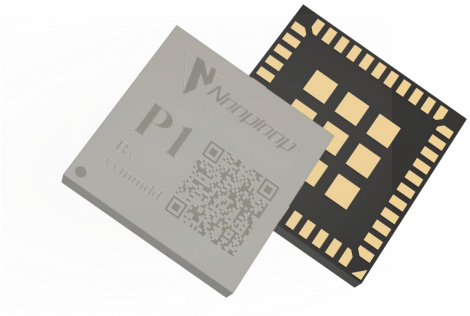
uBeacon Tag UART Protocol

Standard precision UWB module

Version: V1.7



1 Frame

In UART communication, the interaction between the host and the device is governed by a protocol. Data in the protocol is stored in little-endian mode. Additionally, to represent more data using fewer bytes, integers are used to represent floating-point numbers. Therefore, when packaging and unpacking, some data needs to be processed in conjunction with corresponding multipliers.

1.1 Frame Format

A data frame consists of a Frame Header, Payload Size, Payload, and Checksum. The payload contains UID, Frame ID and messages, and frames carrying different messages can achieve different functionalities.

Frame includes Uplink-Frame and Downlink-Frame.

The Uplink-Frame represents data sent by the device.

The Downlink-Frame represents data sent by the host to the tag.

The frame format and composition are as shown in Table 1 to Table 4.

Table 1. Uplink-Frame Format

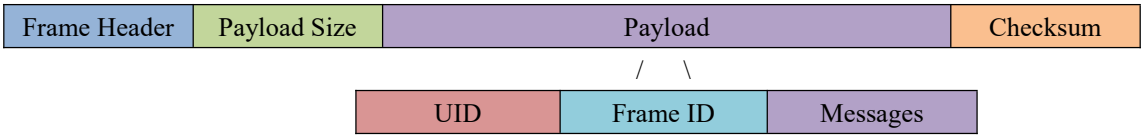


Table 2. Downlink-Frame Format

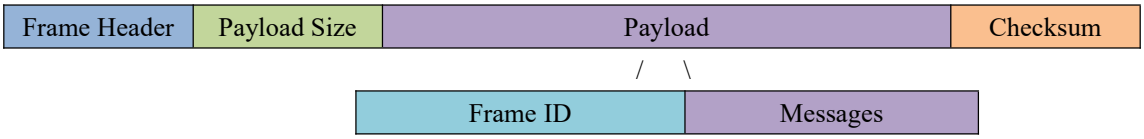


Table 3. Frame Composition

Data		Type	Length (Bytes)	Description
Frame Header		uint8	1	Header of frame, value=0xAA.
Payload Size		uint16	2	Size of payload，max 1005.
Payload	UID	uint8	6	Universally Unique Identifier. Use in Uplink-Frame.
	Frame ID	uint8	1	Frame ID,See Table 4.
	Message	uint8	Payload Size-7	Frame payload data, composed of messages.
Checksum		uint8	1	The Checksum is equal to all previous bytes added.

Table 4. Frame ID List

Frame ID	Value	Description
Serial_Frame_Down	2	Common Downlink-Frame ID.
Serial_Frame_Tag_Up	5	Tag Data Uplink-Frame ID.

1.1.1 Frame Example

The following examples will demonstrate the composition of a data frame.

Delay 10s to Reset:

Host send data: AA 05 00 02 02 02 0A 01 C0.

Table 5. Delay 10s to Reset Frame Example

Data		Hex	Result
Header		AA	Header 0xAA
Payload Size		05 00	5 Bytes
Payload	Frame ID	02	Serial_Frame_Down
	Message	02 02 0A 01	MSG_RESET
Checksum		C0	0xC0

Location Result:

Device send data: AA 2B 00 01 04 02 13 08 C0 05 44 20 56 5A 5C 03 00 00 00 00 2D C1 FB 3F 33 75 92 3F FD 78 99 3F FF FF FC FF 00 00 0B 0E 08 0C 0D 07 02 00 59.

Table 6. Location Result Frame Example

Data		Hex	Result
Header		AA	Header 0xAA
Payload Size		2B 00	43 Bytes
Payload	UID	01 04 02 13 08 C0	
	Frame ID	05	Serial_Frame_Tag_Up
	Message	44 20...02 00	MSG_LOCATION_RESULT
Checksum		59	0x59

1.2 Message Format

A message consists of a Message ID, Need Confirm, Payload Size, and Payload.

Table 7. Message Format

Message ID	Need Confirm	Payload Size	Payload

Table 8. Message Composition

Data	Type	Length	Description
Message ID	uint8	1 Byte	Message ID.
Payload Size	uint8	7 Bits	Size of payload, max 127.
Reserved		1 Bit	Reserved.
Payload	uint8	Payload Size	Message payload data.

1.3 Communication Process

1.3.1 Active Output

Frames actively output by the device. e.g. MSG_LOCATION_RESULT

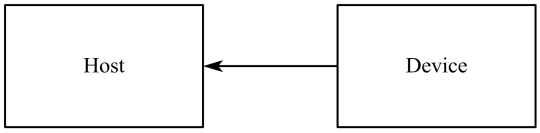


Figure 1. Active Output

1.3.2 Read Communication

The host sends a frame containing a READ_MSG, and the device responds with a frame containing a RESP_MSG.

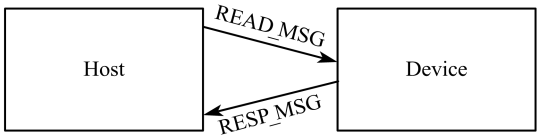


Figure 2. Read Communication

1.3.3 Write Communication

The host sends a frame containing a WRITE_MSG, and the device responds with a frame containing a RESP_MSG.

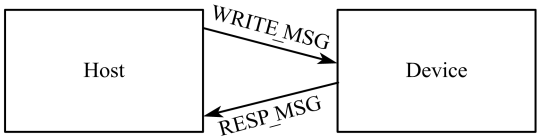


Figure 3. Write Communication

2 Message

Different messages serve different functions, distinguished by their message IDs.

2.1 Message ID

Table 9. Message ID List

Message Group	Message ID	Value	Type
MSG_RESET	MSG_Reset	0x02	WO
MSG_FIND	MSG_Find	0x03	WO
MSG_LOCATION_PARAM	MSG_Read_Location_Param	0x3C	RO
	MSG_Write_Location_Param	0x3D	WO
	MSG_Response_Location_Param	0x3D	RO
MSG_INTERFACE_PARAM	MSG_Read_Interface_Param	0x3E	RO
	MSG_Write_Interface_Param	0x3F	WO
	MSG_Response_Interface_Param	0x3F	RO
MSG_LOCATION_RESULT	MSG_Location_Result	0x44	RO
MSG_HEARTBEAT	MSG_Heartbeat	0x4E	RO
MSG_ANCHOR_SIGNAL	MSG_Anchor_Signal	0x60	RO
MSG_ANCHOR_DDOA	MSG_Anchor_Ddoa	0x61	RO
MSG_RUN_TIME_PARAM	MSG_Read_Run_Time_Param	0x64	RO
	MSG_Write_Run_Time_Param	0x65	WO
	MSG_Response_Location_Param	0x65	RO

2.2 MSG_RESET

Restart current device, usually used to apply changes in parameters.

2.2.1 Composition

Table 10. MSG_RESET Composition

Data	Type	Multiplier	Length (Bytes)	Description
Delay	uint8	1	1	Delay time to reset module, unit s.
Only_restart_when_need	uint8:1	1	1bits	Just restart when needed

only_restart_when_need = true means that the hardware will check if there are any parameter changes that need to be restarted to take effect after receiving this command. If there are, it will restart. If not, this command will ignore it and will not restart

2.2.2 Example

Delay 10s to reset:

Host send message data: 02 02 0A 01.

Table 11. MSG_RESET Parsing Example

Data		Hex	Result
Message ID		02	Value=2 MSG_Reset
Payload Size		02	2
Reserved		0	Reserved
Payload	delay	0A	10s
	only_restart_when_need	01	Need restart

2.3 MSG_FIND

Used to search for devices

2.3.1 Composition

Table 12. MSG_FIND Composition

Data	Type	Multiplier	Length (Bytes)	Description
Duration	uint8	1	1	The duration of continuous external prompts (such as vibration, flashing lights) received by the device

2.3.2 Example

The duration of continuous external prompts (such as vibration, flashing lights) received by the device :

Host send message data: 03 01 0A.

Table 13. MSG_RESET Parsing Example

Data		Hex	Result
Message ID		03	Value=3 MSG_Find
Payload Size		01	1
Reserved		0	Reserved
Payload	duration	0A	10s

2.4 MSG_LOCATION_PARAM

Used for user-defined location parameters.

2.4.1 Composition

Table 14. MSG_LOCATION_PARAM Composition

Data	Type	Multiplier	Length (Bytes)	Description
Reserved	float	1	4	Reserved
Expect_z	float	1	4	User sets the Z-axis value, unit: m.
z_noise	uint8	100	1	User sets the Z-axis noise, unit: m.
Smooth_window	uint8:4	1	4bits	Smooth window,0~5
Max_acceleration[3]	uint8	50	3	3 axis max acceleration{x,y,z},unit:m/s^2.
Output.tag_pos	uint8	1	1bits	Output label positioning result
Output.anchor_packet			1bits	Forwarding beacon packets sent to the backend
Output.anchor_pos			1bits	Output the beacon coordinate information received by the tag
Output.anchor_link_data			1bits	Output link information between beacons
Output.anchor_signal			1bits	Output beacon positioning signal data received by the tag
Output.anchor_ddoa			1bits	Output distance difference information between tags and beacons
Output.tag_pos_even_error			1bits	Output even if the positioning result is invalid
Output.anchor_link_status			1bits	Output the synchronization status of inter beacon links
Sniff_duty_cycle	uint8	1	1	Sniffing periodic

2.4.2 Example

Read location parameters

Host send message data: 3C 00.

Table 15. MSG_LOCATION_PARAM Read Parsing Example

Data	Hex	Result
Message ID	3C	Value=60 MSG_Read_Location_Param
Payload Size	00	0
Reserved	0	Reserved

Device response message data: 3D 0F 00 00 80 3F 9A 99 99 3F 0A 02 0A 0A 01 31 14.

Table 16. MSG_LOCATION_PARAM Response Parsing Example

Data			Hex	Result
Message ID			3D	Value=61 MSG_Response_Location_Param
Payload Size			0F	15 Bytes
Reserved			0	Reserved
Payload	reserved		00 00 80 3F	reserved
	expect_z		9A 99 99 3F	1.2m
	z_noise		0A	0.1m
	smooth_window		02	2
	max_acceleration[3]	x	0A	0.2m/s^2
		y	0A	0.2m/s^2
		z	01	0.02m/s^2
	output		31	Output.tag_pos:1 Output.anchor_signal:1 Output.anchor_ddoa:1
	sniff_duty_cycle		14	20

2.5 MSG_INTERFACE_PARAM

2.5.1 Composition

Table 17. MSG_INTERFACE_PARAM Composition

Data	Type	Multiplier	Length (Bytes)	Description
UART	uint8:1	1	1bit	UART interface switch
IIC	uint8:1	1	1bit	IIC interface switch
UWB	uint8:1	1	1bit	UWB interface switch

2.5.2 Example

Read interface parameters

Host send message data: 3E 00.

Table 18. MSG_INTERFACE_PARAM Read Parsing Example

Data	Hex	Result
Message ID	3E	Value=62 MSG_Read_Interface_Param
Payload Size	00	00
Reserved	0	Reserved

Device response message data: 3F 01 05.

Table 19. MSG_INTERFACE_PARAM Response Parsing Example

Data		Hex	Result
Message ID		3F	Value=63 MSG_Response_Interface_Param
Payload Size		01	1 Byte
Reserved		0	Reserved
Payload	UART	05	1
	IIC		0
	UWB		1

2.6 MSG_LOCATION_RESULT

2.6.1 Composition

Table 20. MSG_LOCATION_RESULT Composition

Data	Type	Multiplier	Length(Bytes)	Description
Location_time	uint64	1	8	Time on module location time.
Pos [3]	float	1	12	3 axis position{x,y,z},unit:m.
Vel [3]	int16	100	6	3 axis velocity{x,y,z},unit:m/s.
Pos_noise [3]	uint8	100	3	3 axis position noise{x,y,z},unit:m.
Vel_noise [3]	uint8	100	3	3 axis velocity noise{x,y,z},unit:m/s.
Map_id	uint8	1	1	The ID of the current map
Error_code	uint8	1	4bits	TBD
Area_id			4bits	The ID of the area

2.6.2 Example

Device active output location result:

Device send message data: 44 22 B5 23 0B 02 00 00 00 00 DD 51 b1 41 EB 58 57 41 AA E7 9A 3F FA FF 04 00 00 00 07 07 04 08 08 04 02 00.

Table 21. MSG_LOCATION_PARAM Response Parsing Example

Data			Hex	Result
Message ID			44	Value=68 MSG_Response_Location_Param
Payload Size			22	34 Bytes
Reserved			0	Reserved
Payload	location_time		B5 23 0b 02 00 00 00 00	34284469 ms
	pos[3]	x	DD 51 b1 41	22.164972 m
		y	EB 58 57 41	13.459208 m
		z	AA E7 9A 3F	1.210195 m
	vel[3]	x	FA FF	-0.06 m/s
		y	04 00	0.04 m/s
		z	00 00	0 m/s
	pos_noise [3]	x	07	0.07 m
		y	07	0.07 m
		z	04	0.04 m
	vel_noise [3]	x	08	0.08 m/s
		y	08	0.08 m/s
		z	04	0.04 m/s
	map_id		02	2
	error_code		00	0
	area_id			0

2.7 MSG_HEARTBEAT

2.7.1 Composition

Table 22. MSG_HEARTBEAT Composition

Data	Type	Multiplier	Length(Bytes)	Description
Battery_percent	uint8	1	7bits	Battery percent , %
Is_charge		1	1bits	Is it charging
Need_restart	uint8	1	1bits	Need to restart
Reset_info_dirty		1	1bits	It is new restart information
Assert_info_dirty		1	1bits	It is new assert information
Restart_cnt		1	3bits	Restart cnt
Hardware_enabled_uart	uint8	1	1bits	Hardware io selection enable interface
Hardware_enabled_iic		1	1bits	
Hardware_enabled_uwb		1	1bits	
Firmware_series	uint8	1	1	Firmware Series
Firmware_version	uint8	1	4	Firmware Version
UID	uint8	1	6	UID

2.7.2 Example

Device active output heartbeat result:

Device send message data: 4E 0E 00 00 00 22 02 00 01 00 01 04 02 13 08 c0

Table 23. MSG_HEARTBEAT Response Parsing Example

Data	Hex	Result
Message ID	4E	Value=78 MSG_HeartBeat
Payload Size	0E	14 Bytes
Reserved	0	Reserved
Battery_percent	00	0%
Is_charge		No
Need_restart	00	No
Reset_info_dirty		No
Assert_info_dirty		No
Restart_cnt		0
Hardware_enabled_uart	00	0
Hardware_enabled_iic		0
Hardware_enabled_uwb		0
Firmware_series	22	
Firmware_version	02 00 01 00	V2.0.1.0
UID	01 04 02 13 08 C0	UID

2.8 MSG_ANCHOR_SIGNAL

Please note that if you need to read data from this register address, you need to enable debug mode on the ubeacon tool and turn on the corresponding data switch in the parameter configuration

2.8.1 Composition

Table 24.MSG_ANCHOR_SIGNAL Composition

Data		Type	Multiplier	Length(Bytes)	Description
Location_time		uint64	1	8	Time on module location time.
Count		uint8	1	4bits	Received beacons num
Area_id				4bits	Area id
Reserved		uint8	1	1	Reserved
Datas[9]	addr	uint16	1	2	Ubeacon addr
	reserved	int16	10	2	Reserved
	reserved	int8	1	1	Reserved
	reserved	uint8	100	1	Reserved
	rx_rssi	uint8	-2	1	Received signal strength
	fp_rssi	uint8	-2	1	First path signal strength
	uwb_clock_offset_ppm	int16	100	2	UWB frequency offset
	mcu_clock_offset_ppm	int16	100	2	RTC frequency offset
	rx_rate	uint8	255	1	Recent packet reception rate of this beacon signal

2.9 MSG_ANCHOR_DDOA

Please note that if you need to read data from this register address, you need to enable debug mode on the ubeacon tool and turn on the corresponding data switch in the parameter configuration

2.9.1 Composition

Table 25.MSG_ANCHOR_DDOA Composition

Data	Type	Multiplier	Length(Bytes)	Description
Location_time	uint64	1	8	Time on module location time.
Addr0	uint16	1	2	Device communication address
Addr1	uint16	1	2	Device communication address
Ddoa	int16	100	2	Distance between beacons
Ddoa_std	uint16	100	2	Standard deviation of distance between beacons

2.9.2 Example

Device active output anchor ddoa:

Device send message data: 61 10 B5 23 0B 02 00 00 00 00 69 11 89 2B 00 00 23 00

Table 26. MSG_ANCHOR_DDOA Response Parsing Example

Data	Hex	Result
Message ID	61	Value=97 MSG_Anchor_Ddoa
Payload Size	10	16 Bytes
Reserved	0	Reserved
Location_time	B5 23 0B 02 00 00 00 00	34284469ms
Addr0	69 11	
Addr1	89 2B	
Ddoa	00 00	0
Ddoa_std	23 00	0.35

2.10 MSG_RUN_TIME_PARAM

2.10.1 Composition

Table 27. MSG_RUN_TIME_PARAM Composition

Data	Type	Multiplier	Length (Bytes)	Description
Sniff_duty_cycle	uint8	1	1	Sniffing periodic

2.10.2 Example

Read run time parameters

Host send message data: 64 00.

Table 28. MSG_RUN_TIME_PARAM Read Parsing Example

Data	Hex	Result
Message ID	64	Value=100 MSG_Read_Run_Time_Param
Payload Size	00	00
Reserved	0	Reserved

Device response message data: 65 01 14.

Table 29. MSG_RUN_TIME_PARAM Response Parsing Example

Data	Hex	Result
Message ID	65	Value=101 MSG_Response_Run_Time_Param
Payload Size	01	1 Byte
Reserved	0	Reserved
Sniff_Duty_Cycle	14	20

3 Document Version

Table 30: Update Log

Version	Date	Description
0.1	20240224	● Initial release.
0.2	20240321	● Fixing some phrasing issues.
1.0	20240506	● Change the document information . ● Official version release.
1.1	20240529	● Update Frame Protocol ,Increase UID and Frame ID.
1.2	20240710	● Change file name to “uBeacon Tag Protocol”
1.3	20240806	● Add MSG_HEARTBEAT
1.4	202408015	● Fixing some mistake
1.5	20240927	● Add MSG_INTERFANCE_PARAM ● Change the UID length to 6 bytes ● Fixing some mistake
1.6	20241028	● Fixing some mistake
1.7	20250506	● Update protocol

.

4 Contact

Company: SZ Noopoop Technology Co.,Ltd.

Address: A2-218, Peihong building, No. 1, Kehui Road, Science Park community, Yuehai street, Nanshan District, Shenzhen

Email: sales@noopoop.com

Tel: 0755-86680090

Website: www.noopoop.com

