IOT-C300 IoT Controller

Communication Protocol
(for LoRaWAN® Version)

Revision History

Date	Doc Version	Description
Mar. 31, 2022	V 1.0	Initial version

Contents

1. Overview

IOT-C300 uses the standard Linovision IoT payload format based on IPSO.

All data are based on following format:

Channel1	Type1	Data1	Channel2	Type2	Data2	Channel 3	...
1 Byte	1 Byte	N Bytes	1 Byte	1 Byte	M Bytes	1 Byte	...

Channel	Description
03	Digital Input 1
04	Digital Input 2
05	Digital Input 3
06	Digital Input 4
07	Digital Output 1
08	Digital Output 2
09	PT100 Input 1
0a	PT100 Input 2
0b	Analog Input (4-20mA) 1
0c	Analog Input (4-20mA) 2
0d	Analog Input (0-10V) 1
0f	Analog Input (0-10V) 2
ff	RS485 Modbus Input/System Info

Note:

1) All explanations and examples in this document are based on HEX format.

2. Uplink Payload

Uplink payloads of C300 are made up of device information and sensor data.

2.1 Device Information

C300 reports basic device information of device every time joining the network.

Channel	Type	Data Size/Byte	Description
ff	01 (Protocol Version)	1	01 => V1
	09 (Hardware Version)	2	01 20 => V1.2
	0a (Software Version)	2	01 01 => V1.1
	0b (Power event)	1	ff => powered on
	16 (Device SN)	8	64 45 B4 34 11 30 00 01 => SN is 64 45 B4 34 11 30 00 01

Example:

ff0bff ff0101 ff166445b43411300001 ff090100 ff0a0101		
Channel	**Type**	**Value**
ff	0b (Power Event)	ff (powered on)
ff	01 (Protocol Version)	01 (V1)
ff	16 (Device SN)	64 45 B4 34 11 30 00 01
ff	09 (Hardware Version)	0100 (V1.0)
ff	0a (Software Version)	0101 (V1.1)

2.2 Sensor Data

C300 reports sensor data according to reporting interval (20min by default). RS232 interface transparently transmits the data fed to it so RS232 uplink doesn't have its own channel or type.

Channel	Type	Byte	Description
03 (DI 1)	00 (Digital Input)	1	00 = low, 01 = high
	c8 (Counter)	4	Unsigned
04 (DI 2)	00 (Digital Input)	1	00 = low, 01 = high
	c8 (Counter)	4	Unsigned
05 (DI 3)	00 (Digital Input)	1	00 = low, 01 = high
	c8 (Counter)	4	Unsigned
06 (DI 4)	00 (Digital Input)	1	00 = low, 01 = high
	c8 (Counter)	4	Unsigned

<table>
<tr><td>07 (DO 1)
08 (DO 2)</td><td>01 (Digital Output)</td><td>1</td><td>00 = low, 01 = high</td></tr>
<tr><td>09 (PT100 1)
0a (PT100 2)</td><td>67 (PT100 Input)</td><td>2</td><td>Collected value × 0.1</td></tr>
<tr><td>0b (4-20mA 1)
0c (4-20mA 2)
0d (0-10V 1)
0e (0-10V 2)</td><td>02 (Analog Input)</td><td>4</td><td>Collected value × 0.01</td></tr>
<tr><td>ff</td><td>19 (RS485)</td><td>Mutable (4-7)</td><td>Total: Byte 1+Byte 2+Byte 3+Value
Byte 1: Channel ID
Byte 2: Data Size
Byte 3: Data Type
<table><tr><th>Code</th><th>Data Type</th></tr><tr><td>00</td><td>Coil</td></tr><tr><td>01</td><td>Discrete</td></tr><tr><td>02</td><td>Input16</td></tr><tr><td>03</td><td>Hold16</td></tr><tr><td>04</td><td>Hold32</td></tr><tr><td>05</td><td>Hold_float</td></tr><tr><td>06</td><td>Input32</td></tr><tr><td>07</td><td>Input_float</td></tr><tr><td>08</td><td>Input_int32_with upper 16 bits</td></tr><tr><td>09</td><td>Input_int32_with lower 16 bits</td></tr><tr><td>0a</td><td>Hold_int32_with upper 16 bits</td></tr><tr><td>0b</td><td>Hold_int32_with lower 16 bits</td></tr></table></td></tr>
<tr><td>ff</td><td>15 (Modbus collecting exception)</td><td>1</td><td>Channel ID of failed Modbus collection.</td></tr>
</table>

Note: Channel ID can be configured in ToolBox.

Channel ID	Description
00	RS485 (Modbus Master) Channel 1
01	RS485 (Modbus Master) Channel 2

02	RS485 (Modbus Master) Channel 3
...	...
0f	RS485 (Modbus Master) Channel 16

Examples:

1. Digital input/counter/output regular uplink

03 c8 16 00 00 00 04 00 00 05 00 00 06 00 01 07 01 00 08 01 01		
Channel	**Type**	**Value**
03 (DI 1)	c8 (Pulse Counter)	16 00 00 00 => 00 00 00 16 = 22
04 (DI 2)	00 (Digital Input)	00 => Low
05 (DI 3)	00 (Digital Input)	00 => Low
06 (DI 4)	00 (Digital Input)	01 => High
07 (DO 1)	01 (Digital Output)	00 => Low
08 (DO 2)	01 (Digital Output)	01 => High

2. RS485 regular uplink

ff 19 07 02 03 15 00					
Channel	**Type**	**Channel ID**	**Data Size**	**Data Type**	**Value**
ff	19 (RS485)	07 => Channel 8	02 => 2 bytes	03 => Hold 16	15 00 => 00 15 = 21

Note: When data type is holding register or input register, ToolBox can set different byte order. Take below Modbus register response from RS485 sensors as example:

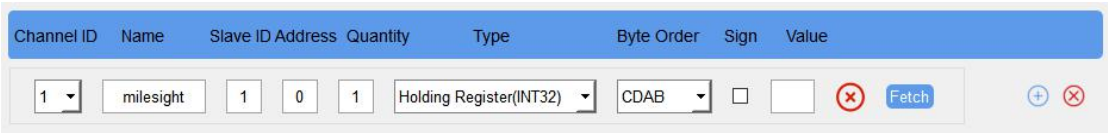
Register Address	Value (Hex)
0	00 15
1	00 20

When using different byte orders, you can use ToolBox to fetch different results and the device will upload data with little endian order.

Data Type	Byte Order	Fetch Result	Uplink (HEX)
Holding/Input Register (INT16)	AB	21 (0x15)	15 00 (BA)
	BA	5376 (0x1500)	00 15 (AB)
Holding/Input Register (INT32)	ABCD	1376288 (0x00150020)	20 00 15 00 (DCBA)
	CDAB	2097173 (0x00200015)	15 00 20 00 (BADC)
	BADC	352329728 (0x15002000)	00 20 00 15 (CDAB)

	DCBA	536876288 (0x20001500)	00 15 00 20 (ABCD)
Holding/Input Register (INT32 with upper 16 bits)	/	21 (0x15)	15 00 00 00
Holding/Input Register (INT32 with lower 16 bits)	/	32 (0x20)	20 00 00 00

If C300 fails to connect the Modbus data, it will an error message.



ff 15 00		
Channel	**Type**	**Value**
ff	15 (Poll Failed)	00 => Channel 1

3. Analog input (4-20mA) regular uplink

0b 02 02 06 00 00 0c 02 00 00 00 00		
Channel	**Type**	**Avg Value**
0b (4-20mA 1)	02 (Analog Input)	02 06 00 00 => 00 00 06 02 = 1538 * 0.01 = 15.38 mA
0c (4-20mA 2)	02 (Analog Input)	00 00 00 00 = 0

4. Analog input (0-10V) regular uplink

0d 02 47 01 00 00 0e 02 00 00 00 00		
Channel	**Type**	**Avg Value**
0d (0-10V 1)	02 (Analog Input)	47 01 00 00 => 00 00 01 47 = 327 * 0.01 = 3.27 V
0e (0-10V 2)		00 00 00 00 = 0

5. PT100 regular uplink

09 67 17 01 0A 67 00 00		
Channel	**Type**	**Avg Value**
09 (PT100 1)	67 (PT100 Input)	17 01 => 01 17 = 279 * 0.1 = 27.9 °C
0a (PT100 2)		00 00 00 00 = 0 °C

3. Downlink Payload

Downlink is used for controlling the UC300 via network server remotely. Downlink port (Application port) is 85 by default and can be configured via ToolBox.
When the channel range is 1~253, the format is:

Channel1	Data1	Reserved	Channel2	Data2	Reserved	Channel3	...
1 Byte	2 Byte	ff	1 Byte	2 Byte	ff	1 Byte	...

When the channel is 255 (ff), the format is:

Channel1	Type1	Data1	Channel2	Type2	Data2	Channel 3	...
1 Byte	1 Byte	N Bytes	1 Byte	1 Byte	M Bytes	1 Byte	...

Examples:

1. DO Control (takes effect only when DO is enabled)

07 01 00 ff		
Channel	**Value**	**Reversed**
07 (Digital Output 1)	01 00 (High)	ff

2. Reporting Interval

ff 03 b0 04		
Channel	**Type**	**Value**
ff	03 (Set Reporting Interval)	b0 04 => 04 b0 = 1200 s = 20 mins

3. Reboot the device

ff 10 ff		
Channel	**Type**	**Reversed**
ff	10 (Reboot)	ff

-END-