UHF
RFID application layer communication protocol

(Version V2.1.1)

Table of contents

1. Overview

UHF RFID The application layer communication protocol is the external communication protocol of the UHF module and reader/writer. Through this protocol , data communication between external devices and read-write modules can be achieved.

1. The UHF module and the host computer use an asynchronous serial interface (UART , TTL level) for data communication . The default serial port baud rate is 115200 , 8 bit data bit, 1 Stop bits, no parity bits, no hardware flow control. Data is transmitted according to a fixed frame format.
2. In addition to serial port communication, the UHF reader/writer also supports TCP/IP communication.

2. Data transmission frame format

The data frame consists of frame header, frame length, CMD type, data, check code and frame tail. As shown in Table 1 .

Data frame transmission format

Frame heade r	Frame length	CMD type	data	BCC code	end of frame
2 byte	2 byte	1 byte	N Charac ter	1 byte	2 byte

2.1. Frame header and frame trailer

The frame header represents the beginning of the data packet, a total of 2 Bytes, which are fixed values 0xC8 , 0x8C or 0xA5 , 0x5A . The end of the frame indicates the end of the data packet, a total of 2 Bytes, which are fixed values 0x0d and 0x0a (carriage return plus line feed) .

2.2. Frame length

The frame length is the length of one frame of data, which is the length of the entire frame of data, including the frame header and tail. The specific calculation formula is:
Frame length = frame header (2 bytes) + frame length (2 bytes) + CMD type (1 byte) + data (N bytes) + BCC code (1 byte) + frame tail (2 bytes) .

2.3. CMD type list

The CMD type is the command type used to distinguish different control commands . The UHF module or reader/writer performs corresponding operations according to the command type.

CMD type list

Command function description	CMD type
Get hardware version number	0x00
Get hardware version number response	0x01
Get firmware version number	0x02
Get firmware version number response	0x03
Get module ID	0x04
Get module ID answer	0x05
Reserve	0x0A~0x0F
Set transmit power	0x10
Set transmit power response	0x11
Get the current device transmit power	0x12
Get the current device transmit power response	0x13
Fixed frequency setting	0x14
Fixed frequency setting response	0x15
Get the current device frequency setting status	0x16
Get the current device fixed frequency setting status response	0x17
SetupGen2 _ parameter	0x20
SetupGen2 _ parameter response	0x21
Get the current Gen2 parameter settings	0x22
Get the current Gen2 Parameter setting response	0x23
CW set up	0x24
CW Set answer	0x25
Reserve	0x26-0x27
Antenna settings	0x28
Antenna Setup Answer	0x29
Get current device antenna settings	0x2A
Get the current device antenna settings response	0x2B
regional settings	0x2C
locale answer	0x2D
Get locale	0x2E
Get locale response	0x2F
Reserve	0x30-0x33
Get the current temperature of the device	0x34

Get the current temperature response of the device	0x35

Reserve	0x36- 0x4D
Get whether the antenna is connected	0x4E
Get the response whether the antenna is connected	0x4F
Reserve	0x50-0x51
Set recommended RF link combination	0x52
Set recommended RF link combination reply	0x53
Get recommended RF Link combination settings	0x54
Get recommended RF Link Combination Setup Answer	0x55
Reserve	0x56-0x5B
Set up FastID function	0x5C
Set FastID function response	0x5D
Get FastID function status	0x5E
Get FastID function status response	0x5F
SetTagFocus _ Function	0x60
SetTagFocus _ functional response	0x61
GetTagFocus _ functional status	0x62
GetTagFocus _ functional status response	0x63
Reserve	0x64-0x67
software reset	0x68
Software reset response	0x69
Reserve	0x6A- 0x6D
Search tag filter settings	0x6E
Tag filter settings response	0x6F
Set inventory EPC+TID or EPC+TID+USER mode	0x70
Set inventory EPC+TID or EPC+TID+USER response	0x71
Get EPC+TID or EPC+TID+USER mode	0x72
Get EPC+TID or EPC+TID+USER mode response	0x73

Reserve	0x74 -0x75
Reserve	0x76~0x7F
Single search for tags	0x80
Single tag seeking response	0x81
Continuous search for tags	0x82
Continuous tag response	0x83
Read data	0x84
Read data response	0x85
write data	0x86
Write data response	0x87
lock tag	0x88
lock tag reply	0x89
Kill Label	0x8A
Kill tag reply	0x8B
Stop searching for tags continuously	0x8C
Stop continuous tag response	0x8D
Authenticate tag	0x8E
Authenticate tag response	0x8F
Reserve	0x90-0x92
Block Write Label	0x93
Block Write tag reply	0x94
Block Erase Label	0x95
Block Erase tag reply	0x96
Set up QT Command parameters	0x97
Set up QT Command parameter response	0x98
Get QT Command parameters	0x99
Get QT Command parameter response	0x9A
QT read operation	0x9B
QT read operation response	0x9C
QT write operation	0x9D
QT write operation response	0x9E
Block Permalock operate	0x9F
Block Permalock Operation response	0xA0
Reserve	0xA1~0xFF

2.4. data

Depending on the CMD type, the data contains data and control information. For command frames, it represents control information, and for response frames, it represents returned data information.

2.5. BCC code

XOR of all bytes of data in each frame (excluding frame header and frame tail).

For example:

0xC8 0x8C 0x00 0x0A 0x43 0x01 0x25 BCC 0x0d 0x0a

BCC = 0x00 ^ 0x0A ^ 0x43 ^ 0x01 ^ 0x25 = 0x6D

3. Communication data frame description

3.1. Device version

3.1.1. Get hardware version number

Data: None
Function: Get hardware version information

Get hardware version command frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x00	none	0x08	0x0D

end of frame
0x0A

Description: This command has no data

Example: Get the card reader hardware version

Command: C8 8C 00 08 00 08 0D 0A

3.1.2. Get hardware version number response

Data: 3 in total bytes , including major version, minor version and supplementary version

Function: respond to hardware version information

Get hardware version response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x0B	0x01	major version	minor version	supplementary version
BCC code	end of frame						
0xxx	0x0D	0x0A					

Description: None

Example: response from a card reader with hardware version number V2.0.1

Command: C8 8C 00 0B 01 02 00 01 09 0D 0A

3.1.3. Get firmware version number

Data: None

Function: Get firmware version information

Get firmware version command frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x02	none	0x0A	0x0D
end of frame							

0x0A

Description: None

Example: Get the card reader firmware version

Command: C8 8C 00 08 02 0A 0D 0A

3.1.4. Get firmware version number response

Data: 3 in total Bytes , including major version, minor version and supplementary version

Function: Respond to firmware version information

Get firmware version response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x0B	0x03	major version	minor version	supplementary version
BCC code		end of frame					
0xxx	0x0D	0x0A					

Description: None

Example: response from a card reader with firmware version number V3.01

Command: C8 8C 00 0B 03 03 00 01 0A 0D 0A

3.1.5. GetGet device **ID**

Data: None

Function: Get module ID

Get module ID command frame

Frame	Frame	CMD	data	BCC	end of

header		length		type		code	frame
0xC8	0x8C	0x00	0x08	0x04	none	0x0C	0x0D
end of frame							
0x0A							

Description: None

Example: Get module ID

Command: C8 8C 00 08 04 0C 0D 0A

3.1.6. Get device **ID** answer

Data: 4 in total Bytes of module ID . Function: Get module ID response.

Get module ID Reply frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x0C	0x05	Dbyte3	DByte2	DByte1
data	BCC code	end of frame					
DByte0	0xxx	0x0D	0x0A				

Example: ID is 0xF1 0xF2 0xF3 Reader response of 0xF4

Command: C8 8C 00 0C 05 F1 F2 F3 F4 0D 0D 0A

3.2. Device parameter settings

3.2.1. Set transmit power

Data: 6 in total bytes , Status It occupies one byte, the antenna number occupies one byte, the read power and the write power each occupy 2 bytes, and the read and write power units are dBm.

Function: Set the read and write power of a specific antenna.

Set transmit power command frame

Frame header	Frame length	CMD type	data

0xA5	0x5A	0x00	0x0E	0x10	Status	Antenna number	Read (MSB)
data			BCC code	end of fra me			
Read (LSB)	Write (MSB)	Write (LSB)	0xxx	0x0D	0x0A		

Status description of each bit

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Rev	Rev	Rev	Rev	Rev	Rev	0 : Do not save 1 : save	Rev

Description: 1 , bit1 is0 Indicates that the current settings will be lost after power off, bit1 is1 It means that the current settings will be saved after power off, and the default power value will be the set value next time when power is turned on. Antenna number in hexadecimal format ; read and write power × 100 Then convert to hexadecimal.

2. The reading power is currently reserved and has no real meaning.

Example: Set the read power of antenna 1 to 0dBm and the write power to 30dBm without saving.

Command: C8 8C 00 0E 10 00 01 00 00 0B B8 AC 0D 0A

3.2.2. Set transmit power response

Data: Set the success flag, success: 0x01 ; failure: 0x00Function: Set whether the transmit power is successful.

Set transmit power response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x11	OK-0x01 Fail-0x00	0xxx	0x0D

end of frame
0x0A

Description: None

Example: Setting transmit power successfully

Command: C8 8C 00 09 11 01 19 0D 0A

3.2.3. Get the current device transmit power

Data: None

Function: Get the current device transmit power.

Get the current device transmit power command frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x12	none	1A	0x0D

end of frame
0x0A

Description: None

Example: Get the current device transmit power

Command: C8 8C 00 08 12 1A 0D 0A

3.2.4. Get the current device transmit power response

Data: Status , antenna number and read and write power of the antenna. The read and write power units are dBm.
Function: Get the read and write power of each antenna of the device.

Get the current device transmit power response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x13	Status	Antenna number	Read (MSB)
data							
Read (LSB)	Write (MSB)	Write (LSB)	Antenna number	Read (MSB)	Read (LSB)	Write (MSB)	Write (LSB)
data						BCC code	end of frame
…	Antenna number	Read (MSB)	Read (LSB)	Write (MSB)	Write (LSB)	0xxx	0x0D
end of frame							
0x0A							

Description: 1. Status, the default is 0x00, reserved for later expansion;

2. For multi-channel devices, the system only sets the power of antenna port 1 by default, and the power of other antenna ports defaults to 0, which needs to be set by the user as needed. An antenna port with power 0 will take the power of antenna 1.

Example: The read power of antenna 1 is 30dBm, the write power is 30dBm, and the power of other antenna ports is 0dB. C8 8C 00 1d 13 00 01 0b b8 0b b8 02 00 00 00 00 03 00 00 00 00 04 00 00 00 00 0a 0d 0a

3.2.5. Fixed frequency setting

Data: Number of fixed frequency points and fixed frequency frequency table.

Function: Set the fixed frequency of the device. Currently, only one frequency is supported.

Fixed frequency setting command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x14	Number of fixed frequency channels	Freq[1] (MSB)	Freq[1]
data					BCC code	end of frame	
Freq[1] (LSB)	…	Freq[n] (MSB)	Freq[n]	Freq[n] (LSB)	0xxx	0x0D	0x0A

Note: The default number of fixed frequency frequency points is 1 , Freq[1] represents the frequency point of fixed frequency. The unit of frequency point Freq is KHz

Example: Set the fixed frequency of 920125KHz (0E0A3D)

Command: C8 8C 00 0C 14 01 0E 0A 3D 20 0D 0A

3.2.6. Fixed frequency setting response

Data: Setting successful: 0x01 ;

Setting failed: 0x00Function:

Frequency hopping setting response.

Fixed frequency setting response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x15	OK : 0x01 fail : 0x00	0xxx	0x0D

end of frame
0x0A

Description: None

Example: Fixed frequency setting successful

Command: C8 8C 00 09 15 01 1D 0D 0A

3.2.7. Get the current device frequency setting status

Data: None

Function: Get the current device fixed frequency status and fixed

frequency table.

Get the current device fixed frequency setting status command frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x16	none	0x1E	0x0D

end of frame
0x0A

Description: None

Example: Get the current device frequency hopping setting status

Command: C8 8C 00 08 16 1E 0D 0A

3.2.8. Get the current device fixed frequency setting status response

Data: Number of fixed frequency points and fixed frequency table
Function: Get the fixed frequency status and fixed frequency table of the device.

Obtain the current device fixed frequency setting response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x17	Number of frequency hopping channels	Freq[1] (MSB)	Freq[1]
data				BCC code	end of frame		
Freq[1] (LSB)	...	Freq[n] (MSB)	Freq[n] (LSB)	0xxx	0x0D	0x0A	

Note: The number of fixed frequency points is 1 , Freq[1] represents the frequency point of the fixed frequency. The unit of frequency point Freq is KHzExample: The fixed frequency point of the device is 920125 (0E0A3D).
C8 8C 00 0C 17 01 0E 0A 3D twenty three 0D 0A

3.2.9. SetupGen2 _ parameter

Data: Session , Q , Coding

Other settings functions: set

gen2 parameter.

Set up gen2 command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x10	0x20	DByte7	DByte6	DByte5
data					BCC code	end of frame	
DByte4	DByte3	DByte2	DByte1	DByte0	0xxx	0x0D	0x0A

Note: The definitions of data bits are as shown in the table below

Data definition instructions

DByte7								DByte6								DByte5								DByte4							
Target			Action			T	Q	StartQ				QQ				MaxQ				D	C		P	Sel				Ses		G	LF		
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
DByte3								DByte2								DByte1								Dbyte0							
Reserv e																															

1. Target _ Settings: select Target of the command parameter

S0	B'000
S1	B'001
S2	B'010
S3	B'011
SL	B'100

2. Action _ Settings: select Action parameters of the command

Action	Matching	Non-Matching
000	assert SL or inventoried →A	de-assert SL or inventoried → *B*
001	assert SL or inventoried → *A*	do nothing
010	do nothing	de-assert SL or inventoried → *B*
011	negate SL or (*A* → *B* , *B* → *A*)	do nothing
100	de-assert SL or inventoried → *B*	de-assert SL or inventoried → *A*
101	de-assert SL or inventoried → *B*	do nothing
110	do nothing	de-assert SL or inventoried → A
111	do nothing	negate SL or (*A* → *B* , *B* → *A*)

3. T _ Settings: Truncate parameter of select command

Disable truncation	B'0
Enable truncation	B'1

4. Q _ set up:

Static Q algorithm	B'0
Dynamic Q algorithm	B'1

Note: Under the fixed Q algorithm, Q is fixed to StartQ , and MinQ and MaxQ are ignored.

5. StartQ _ set up:

0	B'0000	4	B'0100	8	B'1000	12	B'1100
1	B'0001	5	B'0101	9	B'1001	13	B'1101
2	B'0010	6	B'0110	10	B'1010	14	B'1110
3	B'0011	7	B'0111	11	B'1011	15	B'1111

6. MinQ _ set up:

0	B'0000	4	B'0100	8	B'1000	12	B'1100
1	B'0001	5	B'0101	9	B'1001	13	B'1101
2	B'0010	6	B'0110	10	B'1010	14	B'1110
3	B'0011	7	B'0111	11	B'1011	15	B'1111

7. MaxQ _ set up:

0	B'0000	4	B'0100	8	B'1000	12	B'1100
1	B'0001	5	B'0101	9	B'1001	13	B'1101
2	B'0010	6	B'0110	10	B'1010	14	B'1110
3	B'0011	7	B'0111	11	B'1011	15	B'1111

8. D settings : query DR parameters of the command

8	B'0
64/3	B'1

9. Coding _ Settings: query M of command parameter

FM0	B'00
Miller2	B'01
Miller4	B'10
Miller8	B'11

10. P setting : TRext of query command parameter

No pilot	B'0

Use pilot	B'1

11. sel _ Settings: query sel parameter of command

All	B'00
All	B'01
~SL	B'10
SL	B'11

12. ses _ Settings: query The session parameter of the command

S0	B'00
S1	B'01
S2	B'10
S3	B'11

13. G settings : query Target parameters of the command

A	B'0
B	B'1

14 , LFSet (reserved).

Example: Target is set to S1; Action is B'000; Truncate parameter is Disable Truncate; dynamic Q algorithm ; startQ is 4; minQ is 0; maxQ is 15; DR is DR=64/3;The M parameter is Miller4; the TRext parameter is Use pilot; the sel parameter is ALL; Session
The parameter is S1; the Target parameter is A.
Order:

C8 8C 00 10 20 21 40 FD 53 00 00 00 00 FF 0D 0A

3.2.10. SetupGen2 _ parameter response

Data: Setting successful: 0x01 ;
Setting failed: 0x00Function: Set gen2 parameter.

Set up gen2 Parameter response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x21	Ok : 0x01fail : 0x00	0xxx	0x0D
end of							

frame
0x0A

Description: None

Example: Setting gen2 parameters successfully

Command: C8 8C 00 09 21 01 29 0D 0A

3.2.11. Get the current **Gen2** parameter settings

Data: None
Function: Get device gen2 parameter settings.

Get current gen2 Parameter command frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x22	none	0x2A	0x0D

end of frame
0x0A

Description: None

Example: Get the current device gen2 parameter settings

Command: C8 8C 00 08 twenty two 2A 0D 0A

3.2.12. Get the current **Gen2** Parameter setting response

Data: Session , Q , Coding
Other settings functions: Get device Gen2 parameter settings.

Get the current Gen2 Parameter response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x10	0x23	DByte7	DByte6	DByte5
data					BCC	end	

					code	of fra me	
DByte4	DByte3	DByte2	DByte1	DByte0	0xxx	0x0D	0x0A

Note: The data bits are defined as in 3.2.9. Set Gen2 parameters.

Example: Target is set to S1; Action is B'000; Truncate parameter is Disable Truncate; dynamic Q algorithm ; startQ is 4; minQ is 0; maxQ is 15; DR is DR=64/3;The M parameter is Miller4; the TRext parameter is Use pilot; the sel parameter is ALL; Session
The parameter is S1; the Target parameter is A.

Command: C8 8C 00 10 23 21 40 FD 53 00 00 00 00 FC 0D 0A

3.2.13. **CW** set up

Data: CW on : 0x01 ; CW off : 0x00

Function: Turn continuous wave on or off.

CW Set command frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x24	On: 0x01 Off: 0x00	0xxx	0x0D

end of frame
0x0A

Descripti

on: No

example:

open CW

Command: C8 8C 00 09 24 01 2C 0D 0A

3.2.14. **CW** Set answer

Data: Setting successful: 0x01 ;

Setting failed: 0x00Function: Turn on or off continuous wave response.

CW Set response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x25	OK : 0x01 fail : 0x00	0xxx	0x0D

end of frame
0x0A

Description: None

Example: Setup successful

Command: C8 8C 00 09 25 01 2D 0D 0A

3.2.15. Antenna settings

Data: 3 bytes in total, DByte2 byte is saved whether power off; DByte1 and DByte0 bytes total 16

Each bit corresponds to an antenna, and the bit is 1 When, select the corresponding antenna, this bit is 0 , deselect the corresponding antenna. After the antenna is selected, the selected antenna will be automatically rotated during tag inventory.

Function: The single-port module defaults to antenna 1 , and setting other antennas is invalid.

Antenna setting command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x0B	0x28	DByte2	DByte1	DByte0
BCC code		end of frame					
0xxx	0x0D	0x0A					

Note: Dbyte2=0x01 means that the antenna settings are saved when the power is turned off. Dbyte2=0x00 means that the settings are not saved when the power is turned off.

The data bit definitions are as shown in the following table:

Antenna setting data bit definition

DByte 1							
Ant16	Ant15	Ant14	Ant13	Ant12	Ant11	Ant10	Ant9
DByte 0							
Ant8	Ant7	Ant6	Ant5	Ant4	Ant3	Ant2	Ant1

Example: Select antenna No. 2 and antenna No. 14 , and save the settings after power failure.

Command: C8 8C 00 0B 28 01 20 02 01 0D 0A

3.2.16. Antenna Setup Answer

Data: Setting successful: 0x01 ;

Setting failed: 0x00Function: Set the antenna used by the device

Antenna setup response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x29	OK : 0x01 fail : 0x00	0xxx	0x0D
end of frame							
0x0A							

Description: None

Example: Setup successful

Command: C8 8C 00 09 29 01 21 0D 0A

3.2.17. Get current device antenna settings

Data: None

Function: Get the antenna number used by the current device

Get antenna setting command frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x2a	none	0x22	0x0D
end of frame							

0x0A

Description: None

Example: Get the current device antenna settings

Command: C8 8C 00 08 2a 22 0D 0A

3.2.18. Get the current device antenna settings response

Data: 2 bytes in total, 16 bits, each bit corresponds to an antenna. When this bit is 1 , the corresponding antenna is selected.

This bit is 0 , the corresponding antenna is not selected. Function: Get the antenna number used by the current device.

Get antenna setting response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x2b	DByte1	DByte0	0xxx
end of frame							
0x0D	0x0A						

The data bit definitions are as shown in the table:

Antenna setting data bit definition

DByte 1							
Ant16	Ant15	Ant14	Ant13	Ant12	Ant11	Ant10	Ant9
DByte 0							
Ant8	Ant7	Ant6	Ant5	Ant4	Ant3	Ant2	Ant1

Example: Antenna No. 1, Antenna No. 5, Antenna No. 10 and Antenna No. 14 are currently
Command: C8 8C00 0A 2b twenty two 11 12 0D 0A

3.2.19. Band locale

Data: 2 Bytes

function: setting area.

locale command frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x2c	save settings flag	DByte0	0xxx

end of frame	
0x0D	0x0A

Note: When the save settings flag is 0 , the settings are not saved. When it is 1 , the settings are saved. The default is the current region the next time you turn on the computer . Data bit DByte0 The definition is shown in the table

China1	0x01
China2	0x02
Europe	0x04
USA	0x08
Korea	0x16
Japan	0x32
South Africa	0x33
Taiwan	0x34
Vietnam	0x35
Peru	0x36
Russia	0x37
SriLanka	0x38
Azerbaijan	0x39
Iran	0x3A
Malaysia	0x3B
Brazil	0x3C
ETSI_UPPER	0x3D
Australia	0x3E
Indonesia	0x3F

Example: Save the settings and set the area to USA

Command: C8 8C 00 0A 2C 01 08 2F 0D 0A

3.2.20. Band Zone Setup Answer

Data: Setting successful: 0x01 ;

Setting failed: 0x00Function: Set area

locale response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x2D	Ok : 0x01fail : 0x00	0xxx	0x0D

end of frame
0x0A

Description: None

Example: Setup successful

Command: C8 8C 00 09 2D 01 25 0D 0A

3.2.21. Get frequency band area

Data: None
Function: Get the regional settings of the device

Get locale command frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x2E	none	0x26	0x0D
end of frame							
0x0A							

Description: None

Example: Get regional settings

Command: C8 8C 00 08 2E 26 0D 0A

3.2.22. Get band locale response

Data: 2 bytes
Function: Get the regional settings of the device

Get locale reply frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x2f	Ok : 0x01 fail : 0x00	DByte0	0xxx
end of frame							
0x0D	0x0A						

Note: The definition of data bits is as shown in the table

China1	0x01
China2	0x02
Europe	0x04
USA	0x08
Korea	0x16
Japan	0x32
South Africa	0x33
Taiwan	0x34
Vietnam	0x35
Peru	0x36
Russia	0x37
SriLanka	0x38
Azerbaijan	0x39
Iran	0x3A
Malaysia	0x3B
Brazil	0x3C
ETSI_UPPER	0x3D
Australia	0x3E
Indonesia	0x3F

Example: The current device locale is China2

Command: C8 8C 00 0A 01 2F 02 26 0D 0A

3.2.23. Get the current temperature of the device

Data: None

Function: Get the current temperature of the device. The maximum error between this temperature value and the actual temperature value is ±3 ℃ .

Get the current temperature of the device

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x34	none	0x3C	0x0D
end of frame							
0x0A							

Description: None

Example: Get device temperature

Command: C8 8C 00 08 34 3C 0D 0A

3.2.24. Get the current temperature response of the device

Data: Get flag, 0x01 Success, 0x00 fail. Temperature value × 100 , accounting for 2 Bytes, unit is ℃

Function: Get the current temperature response of the device.

Get the current temperature response frame of the device

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x0B	0x35	Ok : 0x01 fail : 0x00	temperature (MSB)	temperature (LSB)
BCC code	end of frame						
0xxx	0x0D	0x0A					

Note : Temperature _

Command: C8 8C 00 0b 35 01 08 98 af 0d 0a

3.2.25. Get the antenna connection status of the device

Data: None
Function: Get the antenna number used by the current device

Get antenna setting command frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x4E	none	0x22	0x0D

end of frame
0x0A

Description: None

Example: Get the current device antenna settings

Command: C8 8C 00 08 4E 46 0D 0A

3.2.26. Get device antenna connection status response

Data: 2 bytes in total, 16 bits, each bit corresponds to a channel, when the bit is 1 , the corresponding channel has a connection line, this bit is 0 , the corresponding channel has no antenna connection. Function: Get the current device antenna connection status.

Get antenna connection

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x4F	DByte1	DByte0	0xxx
end of frame							
0x0D	0x0A						

The data bit definitions are as shown in the table:

Antenna connection data bit definition

DByte 1							
Ant16	Ant15	Ant14	Ant13	Ant12	Ant11	Ant10	Ant9

DByte 0							
Ant8	Ant7	Ant6	Ant5	Ant4	Ant3	Ant2	Ant1

Example: Ports 1, 2, and 4 are connected to antennas

Command: C8 8C 00 0A 4F 00 0B 4E 0D 0A

3.2.27. Set recommended **RF** link combination

Data: three bytes.
Function: Set the recommended RF link combination.

Set recommended RF link combination

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x0B	0x52	Rev	DByte1	DByte0
BCC code		end of frame					
0xxx	0x0D	0x0A					

Description: DByte1 is1 When the power is turned off, the setting is saved and is 0 , the settings will not be saved when the power is turned off. DByte0The settings are as shown in the table below (the default is 0x01, this setting has the best performance).

DByte 0	combination
0x00	DSB_ASK /FM0/ 40 KHz
0x01	PR _ASK /Miller4/ 250KHz
0x02	PR _ASK /Miller4/ 300KHz
0x03	DSB_ASK /FM0/ 400KHz
other	invalid

Example: Set the RF link combination to DSB_ASK / FM0 / 40 KHz, not saved when power off. Command: C8 8C 00 0B 52 00 00 00 59 0D 0A

3.2.28. Set recommended **RF** link combination reply

Data: success or failure flag, 0x01 Success,
0x00 fail. Function: Set recommended RF
Link combination reply.

Set the recommended RF link combination response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x53	Ok : 0x01 fail : 0x00	0xxx	0x0D

end of frame
0x0A

Example: Setup successful

Command: C8 8C 00 09 53 01 5B 0D 0A

3.2.29. Get recommended **RF** Link combination settings

Data: 2 bytes.
Function: Get recommended RF Link combination settings.

Get recommended RF Link combination settings

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x54	Rev	Rev	0xxx
end of frame							
0x0D	0x0A						

Description: None.

Example: Get recommended RF link combination settings

Command: C8 8C 00 0A 54 00 00 5E 0D 0A

3.2.30. Get recommended **RF** Link Combination Setup Answer

Data: three bytes.

Function: Get recommended RF Link combination setup reply.

Get the recommended RF link combination setting response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x0B	0x55	Ok : 0x01 fail : 0x00	Rev	DByte0
BCC code	end of						

	fra me	
0xxx	0x0D	0x0A

Description: DByte0 has the settings shown in the table below

DByte 0	com bina tion
0x00	DSB_ASK /FM0/ 40 KHz
0x01	PR _ASK /Miller4/ 250KHz
0x02	PR _ASK /Miller4/ 300KHz
0x03	DSB_ASK /FM0/ 400KHz
other	inva lid

Example: Current recommendation RF The link combination is DSB_ASK /FM0/ 400KHz Command:

C8 8C 00 0B 55 01 00 03 58 0D 0A

3.2.31. Set **FastID** Function

Data: 2 bytes .
Function: Turn on or off FastID Function.

Set FastID

帧头		帧长度		CMD类型	数据		BCC码
0xC8	0x8C	0x00	0x0A	0x5C	ON：1 OFF：0	Rev	0xxx
帧尾							
0x0D	0x0A						

Description: On: 0x01, Off: 0x00 .

Example: Turn on the FastID function.

Command: C8 8C 00 0A 5C 01 00 57 0D 0A

3.2.32. Set **FastID** functional response

Data: one byte.
Function: Set FastID Function answer.

Set FastID response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x5D	Ok : 0x01 fail : 0x00	0xxx	0x0D
end of frame							
0x0A							

Example: Setup successful

Command: C8 8C 00 09 5D 01 55 0D 0A

3.2.33. Get **FastID** functional status

Data: 2 bytes

Function: Get the current reader FastID Status, whether it is enabled.

Get FastID status

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x5E	Rev	Rev	0xxx
end of frame							
0x0D	0x0A						

Description: None.

Example: Get FastID status.

Command: C8 8C 00 0A 5E 00 00 54 0D 0A

3.2.34. Get **FastID** functional status response

Data: two bytes.
Function: Get FastID status response.

Get FastID status response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x5F	OK : 1 Fail : 0	ON : 1 OFF : 0	0xxx
end of frame							
0x0D	0x0A						

Example: The acquisition is successful and the FastID function is currently on.

Command: C8 8C 00 0A 5F 01 01 55 0D 0A

3.2.35. SetTagfocus _ Function

Data: 2 bytes
TagFocus on or off Function.

SetTagFocus _

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x60	ON : 1 OFF : 0	Rev	0xxx

end of frame	
0x0D	0x0A

Description: On: 0x01, Off: 0x00

. Example: Turn on the

TagFocus function.

Command: C8 8C 00 0A 60 01 00 6B 0D 0A

3.2.36. SetTagFocus _ functional response

Data: one byte.

Function: Set TagFocus Function answer.

SetTagFocus _ Reply frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x61	Ok : 0x01 fail : 0x00	0xxx	0x0D

end of frame	
0x0A	

Example: Setup successful

Command: C8 8C 00 09 61 01 69 0D 0A

3.2.37. GetTagFocus _ functional status

Data: 2 bytes.

Function: Get the current reader TagFocus Status, whether it is enabled.

GetTagFocus _ state

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x62	Rev	Rev	0xxx

end of frame	
0x0D	0x0A

Description: None.

Example: Get TagFocus status.

Command: C8 8C 00 0A 62 00 00 68 0D 0A

3.2.38. GetTagFocus _ functional status response

Data: two bytes.
Function: Get FastID status response.

GetTagFocus _ status response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x63	OK : 1 Fail : 0	ON : 1 OFF : 0	0xxx
end of frame							
0x0D	0x0A						

Example: The acquisition is successful and the TagFocus function is currently on.

Command: C8 8C 00 0A 63 01 01 69 0D 0A

3.2.39. software reset

Data: 0 bytes

Function: Software reset module.

software reset

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x68	none	0x60	0x0D
end of frame							
0x0A							

Description: Send a software reset

command to reset the reader. Example:

Send software reset command.

Command: C8 8C 00 08 68 60 0D 0A

3.2.40. Software reset response

Data: one byte.

Function: Software reset response.

Software reset response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x69	OK : 1 Fail : 0	0xxx	0x0D
end of							

frame
0x0A

Description: None.

Example: Reset successful.

Command: C8 8C 00 09 69 01 61 0D 0A

3.2.41. Search tag filter settings

Data: n bytes.

Function: In the process of searching for tags, the range of tag groups is selected.

Search tag filter settings

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0xxx	0x6E	DByte0	MMB	MSA (MSB)

data							
MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	…	…	…	MData (LSB)
BCC code	end of fra me						
0xxx	0x0D	0x0A					

Description: DByte0 : 0x01 Indicates that the setting value is saved when power is turned off, which is 0x00. Indicates not saving;

MMB : bank for filtering operation Number, 0x01 Indicates EPC , 0x02 Represents TID , 0x03
Indicates USR , other values are illegal values;

MSA : The starting address to start the filtering operation, unit is bit ;

MDL : Filter data length to start filtering operation, unit is bit , 0x00 Indicates no filtering; MData : Data when starting filtering, unit is bytes, if MDL Less than an integer multiple of bytes,The low bits of insufficient bits are filled with 0 .

Example 1: Set tag search filtering rules: TID area filtering, the filtering address is the 0th bit of the TID area , the filtering length is 96 bits, the filtering data is 0xE2003414013301001038D2B5, and the filtering settings are saved when the power is turned off .

Command: C8 8C 00 1A 6E 01 02 00 00 00 60 E2 00 34 14
01 33 01 00 10 38 D2 B5 A9 0D 0A

Example 2: Set tag search filtering rules: no filtering, the filtering settings will be saved when the power is turned off.

Command: C8 8C 00 0E 6E 01 00 00 00 00 00 61
0D 0A

3.2.42. Tag filter settings response

Data: one byte.
Function: Find tag filter settings to answer.

Tag-seeking filter setting response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x6F	OK : 1 Fail : 0	0xxx	0x0D

end of frame
0x0A

Description:

None.

Example:

Setup

successful

Command: C8 8C 00 09 6F 01 67 0D 0A

3.2.43. **EPC** + **TID** or **EPC + USER** Mode settings

Data: 4 bytes
Function: Enable simultaneous reading of EPC+TID or EPC+TID+USER In mode, when the reader continuously searches for tags, it will read the EPC+TID or EPC+USER of the tag at the same time. data.

Mode settings

Frame header	Frame length	CMD type	data

0xC8	0x8C	0x00	0x0C	0x70	Dbyte0	Memory	Address
data	BCC code	end of frame					
Lenth	0xxx	0x0D	0x0A				

Description: Dbyte0 : 0x01 Indicates that the setting value is saved when power is turned off, which is 0x00. Indicates not saving;

Memory : 0x00 , means off; 0x01 , means turn on EPC+TID mode (default address is 0x00, length is 6 words) ;0x02, means turn on EPC+TID+USER mode

Adress : is the starting address of the USER area (unit is word). Lenth : is the length of the USER area (unit is word).

3.2.44. **EPC** + **TID** or **EPC + TID + USER** mode setting answer

Data: 4 bytes.
Function: Read EPC+TID or EPC+TID+USER mode setting response at the same time.

Mode setting response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x71	OK : 1 Fail : 0	0xxx	0x0D
end of frame							
0x0A							

Description:

None.

Example:

Setup successful

Command: C8 8C 00 09 71 01 79 0D 0A

3.2.45. Read **EPC** + **TID** or **EPC + TID + USER** mode status

Data: 2 bytes

Function: Get the current reader and read EPC+TID or EPC+TID+USER at the same time Mode setting status , whether it is enabled.

Read mode setting status

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x72	Rev	Rev	0xxx
end of frame							
0x0D	0x0A						

Description: None.

Example: Get and read the EPC+TID or EPC+TID+USER mode setting status at the same time. Command: C8 8C 00 0A 72 00 00 78 0D 0A

3.2.46. Read **EPC** + **TID** or **EPC + TID + USER** mode status response

Data: 4 bytes.

Function: Get the response of simultaneously reading EPC+TID or EPC+TID+USER mode setting status.

Get mode setting status response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0x0C	0x73	OK : 1 Fail : 0	Memory	Address
data	BCC code	end of frame					
Lenth	0xxx	0x0D	0x0A				

3.3. Label operations

3.3.1. Single inventory label

Data: Timeout (high bit first) timeout time, in milliseconds, if the card search is successful or Time out
Time's up, RFID The module must send back a reply frame.

Function: Search for tags. If a tag is found, only one tag will be returned.

Single inventory label command frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x80	Timeout (MSB)	Timeout (LSB)	0xxx

end of frame	
0x0D	0x0A

Description: None

Example:

looking for tags

Command: C8 8C 00 0A 80 00 64 EE 0D 0A

3.3.2. Single inventory tag response

Data: PC+EPC , RSSI .
Function: Inventory tag response, return tag and reader-related information.

Single inventory tag response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0xxx	0x81	PC (MSB)	PC (LSB)	EPC (MSB)
data							
EPC	EPC	EPC	EPC	EPC	EPC	EPC	EPC
data						BCC code	end of frame
EPC	EPC	EPC (LSB)	RSSI (MSB)	RSSI (LSB)	Ant Num	0xxx	0x0D
end of frame							
0x0A							

Description: RSSI It is expressed in the form of two's complement code, with a total of 16 bits , which is the actual value × 10 . For example , -65.7dBm , then RSSI=FD6F .

Note: EPC The length is PC Decision, this is based on Gen2 protocol, so the frame length is not fixed . at FastID After the function is turned on, if the TID of the tag is read Data, Reply Frame EPC (LSB) 96bit will be added later TID _ data, then RSSI value.

Example: Tag PC=0x3000 , EPC=0xE2003411B802011383258566 response,
RSSI=-65.7dBm , antenna 2 is inventoried.

Command: C8 8C 00 19 81 30 00 E2 00 34 11 B8 02 01 13 83 25 85 66 FD 6F 02 12 0D 0A

3.3.3. Continuous Inventory Label

Data: Number of consecutive inventory tags, 2 in total section.
Function: Continuous inventory label.

Continuous inventory tag command frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x82	Num[1]	Num[0]	0xxx
end of frame							
0x0D	0x0A						

Note: The range of the number of consecutive inventory tags is 1~0xFFFF , and the number is 0 At this time, tags can be inventoried an unlimited number of times .

Note: During the continuous inventory of tags, the reader does not respond to other commands. If you need to execute other commands, It is necessary to send the stop continuous inventory tag command and wait for the stop continuous inventory tag response before sending it. The command to be executed .

Example: The number of consecutive inventory tags is 10000 (0x2710) times

Command: C8 8C 00 0A 82 27 10 BF 0D 0A

3.3.4. Continuous inventory tag response

Data: PC+EPC , RSSI , antenna number.

Function: Inventory tag response, return tag and reader-related information.

Continuous inventory tag response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0xxx	0x83	PC (MSB)	PC (LSB)	EPC (MSB)
data							
EPC	EPC	EPC	EPC	EPC	EPC	EPC	EPC
data						BCC code	end of frame
EPC	EPC	EPC (LSB)	RSSI (MSB)	RSSI (LSB)	Ant Num	0xxx	0x0D
end of frame							
0x0A							

Description: RSSI It is expressed in the form of two's complement

code, with a total of 16 bits , which is the actual value × 10 . For example , -65.7dBm , then RSSI=FD6F .

Note: EPC The length is PC Decision, this is based on Gen2 protocol, so the frame length is not fixed . at FastID After the function or EPC+TID or EPC+USER function is turned on, if the TID of the tag is read data, the TID will be added after the EPC (LSB) of the response frame area or USER area data, and then RSSIvalue.

Example: Tag PC=0x3000, EPC=0xE2003411B802011383258566 response,
RSSI=-65.7dBm, antenna 2 is inventoried.

Command: C8 8C 00 19 83 30 00 E2 00 34 11 B8 02 01 13 83 25 85 66 FD 6F 02 10 0D 0A

3.3.5. Stop continuous inventory label

Data: None
Function: Stop continuous inventory of tags.

Stop continuous inventory tag command

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x08	0x8C	none	0x84	0x0D
end of frame							
0x0A							

Example: Stop continuous inventory of tags

Command: C8 8C 00 08 8C 84 0D 0A

3.3.6. Stop continuous inventory tag response

Data: flag : success: 0x01 ; failure: 0x00

Function: Stop continuous inventory tag response

Stop continuous inventory tag response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x8D	Flag	0xxx	0x0D
end of frame							
0x0A							

Description:

None

Example
:
Success
Command: C8 8C 00 09 8D 01 85 0D 0A

3.3.7. Authenticate tag

Authenticate command function in GS1™ UHF RFID Gen2 v2.0 protocol.

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x8E	AP (MSB)	AP	AP
data							
AP (LSB	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	…	…	…	…	MData	MData (LSB)	D.L.
data							
KeyID	Data (MSB)	…	…	…	…	…	Data (LSB)
BCC code	end of frame						
0xxx	0x0D	0x0A					

AP : 4 -byte access password
MMB : Masked data area (0x00 for Reserve 0x01 For EPC , 0x02 represents TID , 0x03 represents USR)
.
MSA : Masked address. MDL : The length of the mask.
Mdata : mask data.
D.L. : The length of KeyID+Data (unit is bytes), the fixed length is 11. KeyID : KeyID used by the Authenticate command, the default is 0x00, one byte. Data : IChallenge_TAM1 data, fixed at 10 bytes.

3.3.8. Authenticate tag response

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x8F	Flag	Errflag	D.L. (MSB)
data					Check code	end of frame	
D.L. (LSB)	Data (MSB)	…	…	Data (LSB)	0xxx	0x0D	0x0A

Flag: Flag whether reading data is successful,
success : 0x01; failure: 0x00Errflag: error flag, error
flag returned after failure.
DL: The returned data length, in words, is 8 words (16 bytes) if successful; 0 if failed. Data: The specific data returned; when Authenticate fails, there is no Data.

3.3.9. Read tag data area

Data: AP (access password) , MMB , MSA , MDL , MData , memory Bank , S.A.
Starting address (unit: word) , DL The length of data to be read (in words) .
The length of the word is 2 bytes.

Function: Read the data in the data area specified by the tag.

Read data command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x84	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	...	...	...	...	MData	MData (LSB)	MB
data				BCC code	end of frame		
SA (MSB)	SA (LSB)	D.L. (MSB)	D.L. (LSB)	0xxx	0x0D	0x0A	

illustrate:
AP : 4 -byte access password
MMB : Masked data area (0x00 for Reserve 0x01 For EPC , 0x02 represents TID , 0x03 represents USR) .
MSA : The address of the mask (unit is bit). MDL : is the length of the mask (unit is bit). Mdata : mask data.
MB : The data area to be read (0x00 for Reserve 0x01 For EPC , 0x02 represents TID , 0x03 represents USR) .
SA : is the address of the data area to be read (unit is word) .
D.L. : is the length of data to be read (unit: word) .

Example 1: Without filtering, read 3 words of data in the TID area , the starting address is 2, and the access password is 0x55555555

Command: C8 8C 00 16 84 55 55 55 55 00 00 00 00 00 02 00 02 00 03 91 0D 0A

Example 2: TID area filtering, the filtering address is the 2nd bit of the TID area , the filtering length is 13bit , the filtering data is 1110001000000'b, read 6 words of data in the EPC area , the starting address is 2, and the access password is 0x00000000

Command: C8 8C 00 18 84 00 00 00 00 02 00 02 00 0D E2 00 01 00 02 00 06 76 0D 0A

3.3.10. Read tag data area response

Read data response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x85	Flag	Errflag	D.L. (MSB)
data					Check code	end of frame	
D.L. (LSB)	Data (MSB)	…	…	Data (LSB)	0xxx	0x0D	0x0A

Flag : Flag for success in reading data, success: 0x01 ; failure: 0x00Errflag : error flag, error flag returned after failure
DL : The length of the data read, in words Data : The data read

Description: Read data Data It is determined by the data length in the read data command. In addition, if the data reading fails , the response frame will have no Data. Data; read data successfully, error flag Errflag is0x00 . The read operation failed,Errflag Prompt failure type, Errflag is1 , indicating no label; Errflag Ifitis2 , it means the access password is wrong ; Errflag is3 , indicating Read operation failed.

Example: Successfully read 3 words of data in the TID area : 0x123456789ABC

Command: C8 8C 00 12 85 01 00 00 03 12 34 56 78 9A B.C.BB 0D 0A

3.3.11. Write tag data area

Function: Write data to the specified storage area.

Write data command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x86	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	...	...	...	...	MData	MData (LSB)	MB
data							
SA (MSB)	SA (LSB)	D.L. (MSB)	D.L. (LSB)	Data (MSB)	...	...	Data (LSB)
BCC code	end of frame						
0xxx	0x0D	0x0A					

AP : 4 -byte access password

MMB : Masked data area (0x00 for Reserve 0x01 For EPC , 0x02 represents TID , 0x03 represents USR) .

MSA : The address of the mask (unit is bit). MDL : is the length of the mask (unit is bit). Mdata : mask data.

MB : The data area to be written (0x00 for Reserve 0x01 For EPC , 0x02 represents TID , 0x03 represents USR) .

SA : is the address of the data area to be written.

D.L. : is the length of data to be written (unit: word) .

Data : The data to be written, with the high bit first.

Example 1: Without filtering, write 6 words of data in the EPC area , Data=0x00112233445566778899aabb , the starting address is 2 , the

access password is 0x12345678

Command : C8	8C	00	twenty two	86	12	34	56	78	00	00	00	00	00	00	01

00	02	00	06	0	11	t	33	55	77	88	99	AA	BB	A
0D	0A			0		w	44	66						9
						e								
						nt								
						y								
						t								
						w								
						o								

Example 2: TID area filtering, the filtering address is the 0th bit of the TID area , the filtering length is 96bit , the filtering data is 0xE200341401330100103 8D2B5 , write 6 words of data in the EPC area , Data=0x00112233 445566778899aabb, the starting address is 2 , and the access password is 0x00000000

Command: C8	8C	00	2E	86	00	00	00	00	02	00	00	00	60	E2
00	34	14	01	33	01	00	10	38	D2	B5	01	00	02	00	06	00
11	twenty two	33	44	55	66	77	88	99	AA	BB	71	0D	0A			

3.3.12. Write data area response

Data: Flag : Flag of success in writing data, success: 0x01
; failure: 0x00Errflag : error flag, error flag
returned after writing failure

Function: Write data response

Write data response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x87	Flag	Errflag	0xxx
end of frame							
0x0D	0x0A						

Description: Data written successfully, error flag Errflag is 0x00 . Write operation failed, Errflag Prompt failure type, Errflag is 1 , indicating no label; Errflag If it is 2 , it means the access password is wrong; Errflag is 3 , indicating Write operation failed.

Example: Data writing failed, the error flag is 0x03

Command: C8 8C 00 0A 87 00 03 8E 0D 0A

3.3.13. Lock tag

Data: AP (access password) , MMB , MSA , MDL , MData , LD (3 in total bytes)Function: Lock the memory bank of the tag

lock label command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0xxx	0x88	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	...	...	...	...	MData	MData (LSB)	LD (MSB)
data		BCC code	end of frame				
LD	LD (LSB)	0xxx	0x0D	0x0A			

illustrate:
AP : 4 -byte lock password
MMB : Masked data area (0x00 for Reserve 0x01 For EPC , 0x02 represents TID , 0x03 represents USR) .
MSA : The address of the mask (unit is bit). MDL : is the length of the mask (unit is bit). Mdata : mask data.
LD :3 in total byte 24bit , among which, the high 4bit Invalid, bit 0~9 (10 bits in total) isthe Action bit.Bits 10~19 (10 bits in total) are mask bits. For details, please refer to the ISO18000-6C protocol manual.

Example: TID area filtering, the filtering address is the 0th bit of the TID area , the filtering length is 96bit, the filtering data is 0xE2003414013301001038D2B5 , the EPC+RFU area is locked (LD=0x0FC2A0

), the access password is 0x760039AD

Command: C8 8C 00 20 88 76 00 39 AD 02 00 00 00 60 E2

00 34 14 01 33 01 00 10 38 D2 B5 0F C2 A0 FB 0D 0A

3.3.14. **Lock** tag response

Data: flag of whether the lock tag is successful : success: 0x01 ; failure: 0x00

Error flag Errflag : The error flag returned after the lock tag fails. Function: Lock tag response.

Lock tag response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x89	Flag	Errflag	0xxx
end of frame							
0x0D	0x0A						

Description: Lock operation is successful, error flag Errflag is 0x00 . Lock operation failed, Errflag Prompt failure type, Errflag is 1 , indicating no label; Errflag If it is 2 , it means the access password is wrong; Errflag If it is 3 , it means that the Lock operation failed.

Example: Lock successful

Command: C8 8C 00 0A 89 01 00 82 0D 0A

3.3.15. **Kill** Label

Data: KP(kill password) , MMB , MSA , MDL , MDataFunction: kill Label

kill tag command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0x00	0xxx	0x8A	KP (MSB)	KP	KP

data							
KP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							BCC code
MData	…	…	…	…	MData	MData (LSB)	0xxx
end of frame							
0x0D	0x0A						

illustrate:
KP : 4 -byte KILL password
MMB : Masked data area (0x00 for Reserve 0x01 For EPC , 0x02 represents TID , 0x03 represents USR) .
MSA : The address of the mask (unit is bit). MDL : is the length of the mask (unit is bit). Mdata : mask data.
When the value of the tag's KillPwd area is 0x00000000 tag will ignore kill when command, kill command will not succeed

Example: EPC area filtering, the filtering address is the 32nd bit of the EPC area , the filtering length is 96bit , the filtering data is 0x00112233445566778899AABB, and the kill password is 0x760039AD

Command: C8 8C 00 1D 8A 76 00 39 AD 01 00 20 00 60 00
11 22 33 44 55 66 77 88 99 AA BB 34 0D 0A

3.3.16. **Kill** tag reply

Data: kill Flag whether the label is successful : success: 0x01 ; failure: 0x00
Error flag Errflag : kill Error flag function

returned after tag failure: kill tag reply

kill tag response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x8B	Flag	Errflag	0xxx
end of frame							
0x0D	0x0A						

Description: kill Operation successful, error flag Errflag is0x00 . The kill

operation failed, Errflag Prompt failure type, Errflag is1 , indicating no label; Errflag is3 , indicating kill operation failed

Example: kill successful

Command: C8 8C 00 0A 8B 01 00 80 0D 0A

3.3.17. **Block Write** data

Data: AP (access password) , MMB , MSA , MDL , MData , memory Bank , S.A. Starting address (unit: word) , DL Length of data to be written (unit: word) , Data Data to be written

Function: Block Write Data of a specific length to a specific address of the tag.

Block Write command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x93	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	...	...	...	...	MData	MData (LSB)	MB
data							
SA (MSB)	SA (LSB)	D.L. (MSB)	D.L. (LSB)	Data (MSB)	...	...	Data (LSB)
BCC code	end of frame						
0xxx	0x0D	0x0A					

illustrate:
AP : 4 -byte access password
MMB : Masked data area (0x00 for Reserve 0x01 For EPC , 0x02 represents TID , 0x03 represents USR) .
MSA : The address of the mask (unit is bit). MDL : is the length of the mask (unit is bit). Mdata : mask data.
MB : The data area to be written (0x00 for Reserve 0x01 For EPC , 0x02 represents TID , 0x03 represents USR) .
SA : is the address of the data area to be written.
D.L. : is the length of data to be written (unit: word) .
Data : The data to be written, with the high bit first.

Example: Label PC=0x3000, EPC=0xE2003411B802011383258566 , write to EPC area 6

words of data, Data=0x00112233445566778899aabb , the starting address is 2, and the access password is 0x74290fd8

Command: C8 8C 00 2B 93 74 29 0f d8 30 00 E2 00 34 11 B8 02 01 13 83 25 85 66 01 00 02 00 06 00 11 twenty two 33 44 55 66 77 88 99 AA BB 2D 0D 0A

3.3.18. **Block Write** Data response

Data: Block Write Data success flag : success: 0x01 ; failure: 0x00
Error flag Errflag : The error flag returned after the operation fails. Function: Block Write Data response.

Block Write data response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x94	Flag	Errflag	0xxx
end of frame							
0x0D	0x0A						

Description: Operation successful, error flag Errflag is 0x00 . Read data successfully, error flag Errflag is0x00 . Blockwrite Operation failed, Errflag Prompt failure type, Errflag is1 , indicating no label; Errflag If it is 2 , it means the access password is wrong; Errflag is 3 , indicating that Blockwrite is executed The command failed.

Example: Data writing failed, the error flag is 0x01
Command: C8 8C 00 0A 94 00 01 9F 0D 0A

3.3.19. **Block Erase** data

Data: AP (access password) , MMB , MSA , MDL , MData , memory Bank , S.A. Starting address (unit: word) , DL The length to be erased (in words)

Function: Block Erase Specific length to a specific address of the label.

Block Erase command frame

Fra	Frame	CMD	data

me hea der		length		type			
0xC8	0x8C	0xxx	0xxx	0x95	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	…	…	…	…	MData	MData (LSB)	MB
data				BCC code	end of fra me		
SA (MSB)	SA (LSB)	D.L. (MSB)	D.L. (LSB)	0xxx	0x0D	0x0A	

illustrate:
AP : 4 -byte access password
MMB : Masked data area, 0x01 For EPC , 0x02 Represents TID , 0x03 Represents USR . MSA : The address of the mask (unit is bit).
MDL : is the length of the mask
(unit is bit). Mdata : mask data.
MB : is the data area to be written.
SA : is the address of the data area to be written.
D.L. : is the length of data to be written (unit: word) .

Example: Tag PC=0x3000, EPC=0xE2003411B802011383258566, Erase tag EPC Area 6 words of data, starting address is 2, access password is 0x74290fd8

Command: C8 8C 00 1F 95 74 29 0f d8 30 00 E2 00 34 11 B8 02 01 13 83 25 85 66 01 00 02 00 06 1F 0D 0A

3.3.20. **Block Erase** Data response

Data: Block Erase Data success flag : success: 0x01 ; failure: 0x00
Error flag Errflag : The error flag returned
after the operation fails. Function: Block Erase
Data response.

Block Erase data response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x96	Flag	Errflag	0xxx
end of frame							
0x0D	0x0A						

Description: Operation successful, error flag Errflag is 0x00 . Read data successfully, error flag Errflag is 0x00 . Block Erase Operation failed, Errflag Prompt failure type, Errflag is 1 , indicating no label ; Errflag If it is 2 , it means the access password is wrong; Errflag is 3 , indicating execution of Block Erase operation failed .

Example: Data Erase is successful, the error flag is 0x00

Command: C8 8C 00 0A 96 01 00 9D 0D 0A

3.3.21. Set **QT** parameters

Data: AP (access password) , MMB , MSA , MDL , MData , QTDataFunction: Set up QT Command parameters.

Set up QT Command parameters

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x97	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	...	...	...	...	MData	MData (LSB)	QTData
BCC code	end of frame						
0xxx	0x0D	0x0A					

Note: Only supports QT The label of the command can respond to the command.

AP : is the access password of the label.

MMB : is the bank that starts the filtering operation Number, 0x01 Indicates EPC , 0x02 Represents TID , 0x03 Indicates USR , other values are invalid.

MSA : is the starting address of the filtering operation, the unit is bit .

MDL : is the length of filtered data, unit is bit , 0x00 Indicates no filtering.

Mdata : The data when filtering is started, in bytes. If MDL is not an integer multiple of bytes, the low bits are filled with 0 .

QTData : QTData The high 6bit is a reserved bit, bit0 is 0 Indicates no close range control, bit0 is 1 Indicates enabling proximity control;

bit1 is 0 Indicates that the tag enables Private Memory Map,
bit1 is 1 Indicates that the label is enabled
Public Memory Map .

Example: TID area filtering, the filtering address is the 2nd bit of the TID area , the filtering length is 13bit, the filtering data is 1110001000000'b, the access password is 0x00000000, and Private Memory Map is enabled.Enable proximity control.

Command: C8 8C 00 14 97 00 00 00 00 02 00 02 00 0D E2 00 01 6D 0D 0A

3.3.22. Set up **QT** parameter response

Data: one byte.
Function: Set up QT Command parameter response.

Set QT parameter response frame

Frame header		Frame length		CMD type	data	BCC code	end of frame
0xC8	0x8C	0x00	0x09	0x98	OK : 1 Fail : 0	0xxx	0x0D
end of frame							
0x0A							

Description: None.

Example: Setup successful.

Command: C8 8C 00 09 98 01 90 0D 0A

3.3.23. Get **QT** parameters

Data: 0 bytes
Function: Get tag QT Command parameters.

Get QT Command parameters

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x99	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	...	...	...	...	...	MData	MDat a

							(LSB)
BCC code	end of frame						
0xxx	0x0D	0x0A					

Description: Get the tag QT command parameters. Only labels that support QT commands can respond to this command. Example: TID area filtering, the filtering address is the 2nd bit of the TID area , the filtering length is 13bit, and the filtered data is 1110001000000'b, the access password is 0x00000000.

Command: C8 8C 00 12 99 00 00 00 00 02 00 02 00 0D E2 00 64 0D 0A

3.3.24. Get **QT** parameter response

Data: two bytes.
Function: Get QT Command parameter response.

Get QT parameter response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x9A	OK : 1 Fail : 0	DByte0	0xxx
end of frame							
0x0D	0x0A						

Description: DByte0 The high 6bit is a reserved bit, bit0 is 0 Indicates no close range control, bit0 is 1 Indicates enabling proximity control; bit1 is 0 Indicates that the label is enabled Private Memory Map , bit1 is 1 , indicating that the label is Public Memory Map .

Example: Enable Private Memory Map, enable proximity control. Command: C8 8C 00 0A 9A 01 01 90 0D 0A

3.3.25. **QT** Tag read operation

Data: AP (access password) , MMB , MSA , MDL , MData , MB , SA , DL
Features: Via QT Read operation, read tag Private Memory Map After the reading operation is completed , regardless of whether the operation succeeds or fails, the label automatically returns to QT memory before read operation map state.

Set up QT Command parameters

Frame header	Frame length	CMD type	data

der							
0xC8	0x8C	0xxx	0xxx	0x9B	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	…	…	…	MData	MData (LSB)	QTData	MB
data				BCC code	end of fra me		
SA (MSB)	SA (LSB)	D.L. (MSB)	D.L. (LSB)	0xxx	0x0D	0x0A	

Note: Only supports QT The label of the command can respond to the command.

AP : is the access password of the label.

MMB : is the bank that starts the filtering operation Number, 0x01 Indicates EPC , 0x02 Represents TID , 0x03
Indicates USR , other values are invalid.

MSA : is the starting address of the filtering operation, the unit is bit .

MDL : is the length of filtered data, unit is bit , 0x00 Indicates no filtering.

Mdata : The data when filtering is started, in bytes. If MDL is not an integer multiple of bytes, the low bits are filled with 0 .

QTData : High 7bit is a reserved bit, bit0 is0 Indicates no close range control, bit0 is1 Indicates proximity control is enabled.

MB : memory bank , the bank of data that the user needs to write Number

SA : is the starting address of the data to be written, unit is word.

D.L. : is the length of data to be written, in words.

Example: TID area filtering, the filtering address is the 2nd bit of the TID area , the filtering length is 13bit , the filtering data is 1110001000000'b, read 6 words of data in the EPC area , the starting address is 2, the access password is 0x00000000, close reading operate

Command: C8 8C 00 19 9B 00 00 00 00 02 00 02 00 0D E2 00 01 01 00 02 00 06 69 0D 0A

3.3.26. QT Tag read operation response

Data: Flag : Flag of success in reading data, success:
0x01 ; failure: 0x00 Errflag : Error flag, error flag returned after failure to read data. DL : Read data length, unit is word
Data : read data

Function: QT Tag read operation response.

QT tag read operation response frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x9C	Flag	Errflag	D.L. (MSB)
data					BCC code	end of frame	
D.L. (LSB)	Data (MSB)	…	…	Data (LSB)	0xxx	0x0D	0x0A

Description: Read data Data It is determined by the data length in the read data command. In addition, if the data reading fails , the response frame will have no Data. Data; read data successfully, error flag Errflag is0x00 . The read operation failed,Errflag Prompt failure type, Errflag is1 , indicating no label; Errflag Ifitis2 , it means the access password is wrong; Errflag Ifitis3 , it means the operation failed.

Example: Successfully read 3 words of data in the TID area:

0x123456789abc

Command: C8 8C 00 12 9C 01 00 00 03 12 34 56 78 9A B.C. A2 0D 0A

3.3.27. QT tag write operation

Data: AP (access password) , MMB , MSA , MDL , MData , memory Bank , S.A. Starting address (unit: word) , DL Length of data to be written (unit: word) , Data Data to be written

Features: Via QT Write operation, write data of a specific length to the tag's Private Memory Map , At a specific address, after the write operation is completed, regardless of whether the operation succeeds or fails, the label automatically returns to the memory map state before the QT read operation .

QT tag write operation command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x9D	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	...	...	...	MData	MData	QTData	MB
data							
SA (MSB)	SA (LSB)	D.L. (MSB)	D.L. (LSB)	Data (MSB)	...	...	Data (LSB)
BCC code	end of frame						
0xxx	0x0D	0x0A					

Note: Only supports QT The label of the command can respond to the command.

AP : is the access password of the label.

MMB : is the bank that starts the filtering operation Number, 0x01 Indicates EPC , 0x02 Represents TID , 0x03 Indicates USR , other values are invalid.

MSA : is the starting address of the filtering operation, the unit is bit .

MDL : is the length of filtered data, unit is bit , 0x00 Indicates no filtering.

Mdata : The data when filtering is started, in bytes. If MDL is not an integer multiple of bytes, the low bits are filled with 0 .

QTData : High 7bit is a reserved bit, bit0 is0 Indicates no close range control, bit0 is1 Indicates proximity control is enabled.

MB : memory bank , the bank of data that the user needs to write Number

SA : is the starting address of the data to be written, unit is word.

D.L. : is the length of data to be written, in words.

Data : The data to be written, with the high-order bit first.

Example: TID area filtering, the filtering address is the 0th bit of the TID area , the filtering length is 96bit , the filtering data is 0xE2003414013301001038D2B5 , write 6 words of data in the EPC area , Data=0x00112233
445566778899aabb , starting address is 2 , access password is 0x00000000, close write operation

Comman 8C 00 2F 9D 00 00 00 00 02 00 00 00 60 E2
d: C8
00 34 14 01 33 01 00 10 38 D2 B5 01 01 00 02 00 06 00 11
twenty two 33 77 88 99 AA BB 6A 0D 0A
44 55 66

3.3.28. QT tag write operation response

Data: flag indicating whether writing data is successful : success: 0x01 ; failure: 0x00

Error flag Errflag : The error flag returned after failure to write data. Function: Write data response.

QT Tag write operation response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0x0A	0x9E	Flag	Errflag	0xxx
end of frame							
0x0D	0x0A						

Description: Data written successfully, error flag Errflag is 0x00 . Write operation failed, Errflag Prompt failure type, Errflag is 1 , indicating no label; Errflag If it is 2 , it means the access password is wrong; Errflag is 3 , indicating Write operation failed.

Example: Data writing failed, the error flag is 0x03

Command: C8 8C 00 0A 9E 00 03 97 0D 0A

3.3.29. **Block Permalock** operate

Data: AP (access password) , MMB , MSA , MDL , MData , ReadLock , MemBank ,
BlockPtr , BlockRange , Mask .

Function: BlockPermalock
operate.

Block Permalock Operation command frame

Frame header		Frame length		CMD type	data		
0xC8	0x8C	0xxx	0xxx	0x9F	AP (MSB)	AP	AP
data							
AP (LSB)	MMB	MSA (MSB)	MSA (LSB)	MDL (MSB)	MDL (LSB)	MData (MSB)	MData
data							
MData	...	...	...	MData	MData (LSB)	ReadLock	MB
data							
BlockPtr (MSB)	BlockPtr (LSB)	BlockRange (M.S.B)	BlockRange (LSB)	Mask (MSB)	...	...	Mask (LSB)
BCC code	end of frame						
0xxx	0x0D	0x0A					

Note: Only supports Block Permalock The label of the command can respond to the command.

AP : Access password of the tag.

MMB : The bank that starts the filtering operation number,0x01 is EPC , 0x02 is TID , 0x03 is USER area,Other values are not valid.

MSA : The starting address to start filtering operation, unit is bit.

MDL : Filter data length to start filtering operation, unit is bit , 0x00 Indicates no filtering.

Mdata : The data when filtering is started, in bytes. If MDL is not an integer multiple of bytes, the low bits are filled with 0 .

ReadLock : The high 7 bits are reserved bits, bit0 is 0 Indicates Read , bit0 is 1 Represents Permalock . MB : for memory bank , the data area to be operated on.

BlockPtr : for Block Starting address, unit is 16 blocks, a block is 8 bytes .

BlockRange : block Range, unit is 16 blocks _

Mask : block mask data, high-order bit first, two bytes of 16 bits corresponding to whether 16 blocks are selected.

Example: TID area filtering, the filtering address is the 0th bit of the TID area , the filtering length is 96bit, the filtered data is 0xE200341401330100103 8D2B5, readlock=0, MB=3, BlockPtr=0,BlockRange=1 , access password is 0x00000000

Command: C8	8C	0	twenty three	9F	0	0	00	00	02	00	00	00	60	E2

00 34 14 01 33 01 00 10 38 D2 B5 00 03 00 00 00 01 62 0D 0A

3.3.30. **Block Permalock** Operation response

Data: Flag Is the success flag, success: 0x01 ; failure: 0x00 ;
Errflag is the error flag, the error flag returned after the operation fails. Function: Block Permalock operation response.

Block Permalock Operation response frame

Frame header		Frame length		CMD type	data		BCC code
0xC8	0x8C	0x00	0xxx	0xA0	Flag	Errflag	0xxx
end of frame							
0x0D	0x0A						

Description: Block Permalock Success, error flag Errflag is 0x00 . Block Permalock Failure, Errflag Prompt failure type, Errflag is 1 , indicating no label; Errflag If it is 2 , it means the access password is

wrong; Errflag is3 , indicating Block Permalock operation failed.

If Block Permalock readlock in command The parameter is 0 , Errflag There is a corresponding data response, and the data length is BlockRange words.

Example: Block Permalock Success, Readlock=0, BlockRange=1, data is 0xF000Command: C8 8C 00 0C A0 01 00 F0 00 5D 0D 0A