Unit RFID-UHF

Common Control Commands

目录

1、Firmware Command Overview

1.1 Command Frame Format

The firmware command consists of a frame header, frame type, command code, command data length, command parameters, checksum, and frame tail, all represented in hexadecimal. For example:

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	00	07	00	01	01	09	7E

Header: 0xBB
Type: 0x00
Command: 0x07
Parameter Length: 0x0001
Parameter: 0x01
Checksum: 0x09
End: 0x7E

The checksum is the cumulative sum from Type to the last Parameter, with only the least significant byte (LSB) of the sum retained.

1.2 Command Type

Type	Description
0x00	Command Frame: sent from the host computer to the M100 chip
0x01	Response Frame: sent from the M100 chip back to the host computer
0x02	Notification Frame: sent from the M100 chip back to the host computer

Each command frame has a corresponding response frame, which indicates whether the command has been executed.

Single polling commands and multiple polling commands also have their corresponding Notification Frames. The number of Notification Frames sent is determined by the MCU based on the reading status and is sent autonomously to the host computer. When the reader detects a single tag, it sends one Notification Frame; when it detects multiple tags, it sends multiple Notification Frames.

2、Common Command Definitions

2.1 Get Reader Module Information

2.1.1 Command Frame

Type: 0x00
Command: 0x03
Parameter: 0x00-Hardware Version 0x01-Software Version 0x02-Manufacturer

- Hardware Version（00）:

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	00	03	00	01	00	04	7E

- Software Version（01）:

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	00	03	00	01	01	05	7E

- Manufacturer（03）:

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	00	03	00	01	02	06	7E

2.1.2 Notification Frame

Type: 0x01
Command: 0x03
Parameter: 0x00-Hardware Version 0x01-Software Version 0x02-Manufacturer
Information: ASCII

- Hardware Version:

例: M100 V1.00——ASCII: 4D 31 30 30 20 56 31 2E 30 30

Header	Type	Command	PL（MSB）	PL（LSB）	信息 Type	版本信息	Checksum	End
BB	01	03	00	0B	00	见下表	22	7E

版本信息:

M	1	0	0		V	1	.	0	0
4D	31	30	30	20	56	31	2E	30	30

- Software Version:

Similarly, hardware version information.

- Manufacturer:

Similarly, hardware version information.

2.2 Single Polling Command

2.2.1 Command Frame

Type: 0x00
Command: 0x22
Parameter Length: 0x0000

Checksum: 0x22

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	22	00	00	22	7E

2.2.2 Notification Frame

After the chip receives the single polling command, if it can read a tag with a correct CRC check, the chip MCU will return data containing RSSI, PC, EPC, and CRC. When one tag EPC is read, a single command response is returned; when multiple tags are read, multiple command responses are returned. As follows:

Type: 0x02

Command: 0x22

Parameter Length: 0x0011

RSSI: 0xC9

PC: 0x3400

EPC: 0x30751FEB705C5904E3D50D70

CRC: 0x3A76

Checksum: 0xEF

Header	Type	Command	PL（MSB）	PL（LSB）	RSSI	PC（MSB）	PC（LSB）
BB	00	22	00	11	C9	34	00
EPC（MSB）							
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）			CRC（MSB）		CRC（LSB）		Checksum		End	
70			3A		76		EF		7E	

The RSSI value reflects the signal level at the chip input and does not include antenna gain or directional coupler attenuation. RSSI represents the signal strength at the chip input; it is a signed hexadecimal value with the unit dBm. In the above example, the RSSI is 0xC9, indicating a signal strength of –55 dBm at the chip input.

2.2.3 Notification Frame

If no tag is received or the returned data fails the CRC check, the error code 0x15 will be returned, as shown below:

Type: 0x01

Command: 0xFF

Parameter Length: 0x01

Parameter: 0x15

Checksum: 0x16

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	15	16	7E

2.3 Multiple Polling Command

2.3.1 Command Frame

The polling count is limited to 0 – 65535 times. If the polling count is 10,000, the command is as follows:

Type: 0x00

Command: 0x27

Parameter Length: 0x0003

Reserved: 0x22

Polling Count: 0x2710

Checksum: 0x22

Header	Type	Command	PL（MSB）	PL（LSB）	Reserved	CNT（MSB）	CNT（LSB）	Checksum	End
BB	00	27	00	03	22	27	10	83	7E

2.3.2 Notification Frame

After the chip receives the multiple polling command, if it can read a tag with a correct CRC check, the chip MCU will return data containing RSSI, PC, EPC, and CRC. When one tag EPC is read, a single command response is returned; when multiple tags are read, multiple command responses are returned. As follows:

Type: 0x02

Command: 0x27

Parameter Length: 0x0011

RSSI: 0xC9

PC: 0x3400

EPC: 0x30751FEB705C5904E3D50D70

CRC: 0x3A76

Checksum: 0xEF

Header	Type	Command	PL（MSB）	PL（LSB）	RSSI	PC（MSB）	PC（LSB）
BB	02	22	00	11	C9	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	CRC（MSB）	CRC（LSB）	Checksum	End
70	3A	76	EF	7E

2.3.3 Notification Frame

If no tag is received or the returned data fails the CRC check, the error code 0x15 will be returned, as shown below:

Type: 0x01

Command: 0xFF

Parameter Length: 0x01

Parameter: 0x15

Checksum: 0x16

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	15	16	7E

2.4 Stop Multiple Polling Command

2.4.1 Command Frame

Immediately stop the multiple polling operation; this is not a pause operation.

Type: 0x00

Command: 0x28

Parameter Length: 0x0000

Checksum: 0x28

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	28	00	00	28	7E

2.4.2 Notification Frame

If the Stop Multiple Polling Command is executed successfully, the firmware will return the following response:

Type: 0x01

Command: 0x28

Parameter Length: 0x0001

Parameter: 0x00

Checksum: 0x2A

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	28	00	01	00	2A	7E

2.5 Set Select Parameter Command

2.5.1 Command Frame

Set the Select parameters and simultaneously set the Select mode to 0x02 (send the Select command before any polling operations on the tags). In a multi-tag scenario, the Select parameters can be used to poll, read, write, and perform other operations only on specific tags. For example:

Type: 0x00

Command: 0x0C

Parameter Length: 0x0013

SelParam: 0x01 (Target: 3'b000, Action: 3'b000, MemBank: 2'b01)

Ptr: 0x00000020 (in bits, not words) Starting from the EPC storage bit
Mask Length: 0x60 (6 words，96bits)
Truncate: 0x00 (0x00 is Disable truncation，0x80 is Enable truncation)
Mask: 0x30751FEB705C5904E3D50D70
Checksum: 0xAD

Header	Type	Command	PL（MSB）	PL（LSB）	SelParam
BB	00	0C	00	13	01

Ptr（MSB）			Ptr（LSB）	MaskLen	Truncate
00	00	00	20	60	00

Mask（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

Mask（LSB）	Checksum	End
70	AD	7E

SelParam occupies 1 byte in total, where Target takes the highest 3 bits, Action occupies the middle 3 bits, and MemBank occupies the lowest 2 bits.

MemBank meanings are as follows:

2'b00: Tag RFU memory bank

2'b01: Tag EPC memory bank

2'b10: Tag TID memory bank

2'b11: Tag User memory bank

For detailed meanings of Target and Action, please refer to the EPC Gen2 protocol.

When the Select Mask length is greater than 80 bits (5 words), sending the Select command will first set all tags in the field to Inventoried Flag = A and SL Flag = ~SL, and then perform the operation according to the chosen Action.When the Select Mask length is less than 80 bits (5 words), the tag state is not preset to Inventoried Flag = A and SL Flag = ~SL through the Select command.

2.5.2 Notification Frame

When the Select parameters are successfully set, the firmware returns as follows:

Type: 0x01
Command: 0x0C
Parameter Length: 0x0001
Data: 0x00
Checksum: 0x0E

Header	Type	Command	PL（MSB）	PL（LSB）	Data	Checksum	End
BB	01	0C	00	01	00	0E	7E

2.6 Get Select Parameter

2.6.1 Command Frame

Type: 0x00
Command: 0x0B
Parameter Length: 0x0000

Checksum: 0x0B

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	0B	00	00	0B	7E

2.6.2 Notification Frame

Type: 0x01
Command: 0x0B
Parameter Length: 0x0013
SelParam: 0x01 (Target: 3'b000, Action: 3'b000, MemBank: 2'b01)
Ptr: 0x00000020 (in bits, not words) Starting from the EPC storage bit
Mask Length: 0x60 (6 words，96bits)
Truncate: 0x00 (0x00 is Disable truncation，0x80 is Enable truncation)
Mask: 0x30751FEB705C5904E3D50D70
Checksum: 0xAD

Header	Type	Command	PL（MSB）	PL（LSB）	SelParam
BB	01	0B	00	13	01

Ptr（MSB）			Ptr（LSB）	MaskLen	Truncate
00	00	00	20	60	00

Mask（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

Mask（LSB）	Checksum	End
70	AD	7E

2.7 Set Select Mode

2.7.1 Command Frame

If the Select parameter has already been set, executing this command enables the Select mode. For example, to cancel the Select command:

Type: 0x00
Command: 0x12
Parameter Length: 0x0001
Select Mode: 0x01
Checksum: 0x14

Header	Type	Command	PL（MSB）	PL（LSB）	Select Mode	Checksum	End
BB	00	12	00	01	01	14	7E

Meaning of Select Mode:

0x00: Send the Select command in advance before every tag operation to select a specific tag.

0x01: Do not send the Select command before tag operations.

0x02: Send the Select command only before tag operations other than the polling Inventory, such as in Read, Write, Lock, or Kill operations, first using Select to choose a specific tag.

2.7.2 Notification Frame

After successfully configuring to cancel or send the Select command, the firmware returns as follows:

Type: 0x01
Command: 0x0C
Parameter Length: 0x0001
Data: 0x00 (Execution successful)
Checksum: 0x0E

Header	Type	Command	PL（MSB）	PL（LSB）	Data	Checksum	End
BB	01	0C	00	01	00	0E	7E

2.8 Read Tag Memory Area

2.8.1 Command Frame

For a single tag, read the data located at the specified address and length in the tag's Memory Bank. The starting address offset (SA) and data length (DL) for reading the tag memory area are measured in words, meaning 2 bytes / 16 bits each. You should set the Select parameter beforehand to select the target tag before performing the Read Tag Memory operation. If the Access Password is all zeros, the Access command will not be sent.

Type: 0x00
Command: 0x39
Parameter Length: 0x0009
Access Password: 0x0000FFFF
MemBank: 0x03 (User area)
Read Tag Memory Area Address Offset-SA: 0x0000
Read Tag Memory Area Length-DL: 0x0002
Checksum: 0x45

Header	Type	Command	PL（MSB）	PL（LSB）	AP（MSB）		
BB	00	39	00	09	00	00	FF

AP（LSB）	MemBank	SA（MSB）	SA（LSB）	DL（MSB）	DL（LSB）	Checksum	End
FF	03	00	00	00	02	45	7E

2.8.2 Notification Frame

Type: 0x01
Command: 0x39
Parameter Length: 0x0013
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Data: 0x12345678
Checksum: 0xB0

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	39	00	13	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Data（MSB）			Data（LSB）	Checksum	End
70	12	34	56	78	B0	7E

If the tag is not within the field or the specified EPC code is incorrect, error code 0x09 will be returned as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x09
Checksum: 0x0A

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	09	0A	7E

If the Access Password is incorrect, error code 0x16 will be returned, along with the PC+EPC of the tag being operated, as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0x16
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Checksum	End
70	75	7E

If the operated tag returns an error code defined by the EPC Gen2 protocol, note that only the lower 4 bits of EPC Gen2 error codes are valid. The Notification Frame will therefore OR the tag-returned error code with 0xA0 before sending it back. For instance, if the address offset or data length specified in the Parameter field is incorrect, or the requested read length exceeds the tag's memory area, the tag will return EPC Gen2 error code 0x03 (Memory Overrun). Consequently, the Notification Frame will return error code 0xA3 along with the PC+EPC of the tag being operated, as shown below:

Type: 0x01
Command: 0xFF

Parameter Length: 0x0010
Parameter: 0xA3
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x02

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	A3	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Checksum	End
70	02	7E

2.9 Write Tag Memory Area

2.9.1 Command Frame

For a single tag, write data to the specified address and length in the tag's Memory Bank. The starting address offset (SA) and the data length to be written (DL) are measured in words, meaning 2 bytes / 16 bits each. You should set the Select parameter beforehand to choose the target tag for the Write Tag Memory operation. If the Access Password is all zeros, the Access command will not be sent. The data length (DT) to be written into the tag's memory area must not exceed 32 words, i.e., 64 bytes / 512 bits.

Type: 0x00
Command: 0x49
Parameter Length: 0x000D
Access Password: 0x0000FFFF
MemBank: 0x03
Read Tag Memory Area Address Offset-SA: 0x0000
Read Tag Memory Area Length-DL: 0x0002
DT: 0x12345678
Checksum: 0x6D

Header	Type	Command	PL（MSB）	PL（LSB）	AP（MSB）		
BB	00	49	00	0D	00	00	FF

AP（LSB）	MemBank	SA（MSB）	SA（LSB）	DL（MSB）	DL（LSB）
FF	03	00	00	00	02

DT（MSB）			DT（LSB）	Checksum	End
12	34	56	78	6D	7E

2.9.2 Notification Frame

After writing data to the tag memory area, if the reader chip receives a correct return value from the tag, the Notification Frame is as follows:

Type: 0x01

Command: 0x39

Parameter Length: 0x0010

URL Length: 0x0E

PC: 0x3400

EPC: 0x30751FEB705C5904E3D50D70

Parameter: 0x00 (Execution successful)

Checksum: 0xA9

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	49	00	10	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Parameter	Checksum	End
70	00	A9	7E

If the tag is not within the field or the specified EPC code is incorrect, error code 0x10 will be returned as follows:

Type: 0x01

Command: 0xFF

Parameter Length: 0x0001

Parameter: 0x10

Checksum: 0x0A

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	10	0A	7E

If the Access Password is incorrect, error code 0x16 will be returned, along with the PC+EPC of the tag being operated, as follows:

Type: 0x01

Command: 0xFF

Parameter Length: 0x0010

Parameter: 0x16

URL Length: 0x0E

PC: 0x3400

EPC: 0x30751FEB705C5904E3D50D70

Checksum: 0x75

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Checksum	End
70	75	7E

If the operated tag returns an error code defined by the EPC Gen2 protocol, note that only the lower 4 bits of EPC Gen2 error codes are valid. Consequently, the Notification

Frame will OR the tag-returned error code with 0xB0 before sending it back. For example, if the address offset or data length specified in the Parameter field is incorrect, or the requested read length exceeds the tag's memory area, the tag will return EPC Gen2 error code 0x03 (Memory Overrun). In this case, the Notification Frame will return error code 0xB3 along with the PC+EPC of the tag being operated, as shown below:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0xB3
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x12

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	B3	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Checksum	End
70	12	7E

2.10 Lock the Lock Tag Data Storage Area

2.10.1 Command Frame

For a single tag, lock or unlock the tag's memory area. This command should be preceded by setting the Select parameter to choose the specific tag for the Lock operation. For example, to lock the Access Password, use the following command:

Type: 0x00
Command: 0x82
Parameter Length: 0x0007
Access Password: 0x0000FFFF
Lock Operation-LD: 0x020080
Checksum: 0x09

Header	Type	Command	PL（MSB）	PL（LSB）	AP（MSB）		
BB	00	82	00	07	00	00	FF

AP（LSB）	LD（MSB）		LD（LSB）	Checksum	End
FF	02	00	80	09	7E

LD's upper 4 bits are Reserved, while the remaining 20 bits constitute the Lock operation payload, which consists of Mask and Action—10 bits each, ordered from MSB to LSB. For detailed information, refer to EPC Gen2 standard v1.2.0, Section 6.3.2.11.3.5.

Mask serves as a bit-mask; only actions whose corresponding Mask bit is 1 are valid. Each memory area's Action occupies 2 bits (00 – 11), representing in sequence: Open,

Permanently Open, Lock, Permanently Lock.

For example, if the Kill Mask is 2 bits 00, the Kill Action is ignored regardless of its value. When the Kill Mask is 2 bits 10 and the Kill Action is 2 bits 10, the Kill Password is locked (non-permanent lock) and can be read or written only with a valid Access Password.

The meaning of each Mask and Action bit is shown below.

Lock-Command Payload

19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Kill Mask		Access Mask		EPC Mask		TID Mask		User Mask		Kill Action		Access Action		EPC Action		TID Action		User Action	

Masks and Associated Action Fields

	Kill pwd		Access pwd		EPC memory		TID memory		User memory	
	19	18	17	16	15	14	13	12	11	10
Mask	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write
	9	8	7	6	5	4	3	2	1	0
Action	pwd read/ write	perma lock	pwd read/ write	perma lock	pwd write	perma lock	pwd write	perma lock	pwd write	perma lock

pwd-write	permalock	Description
0	0	Associated memory bank is writeable from either the **open** or **secured** states.
0	1	Associated memory bank is permanently writeable from either the **open** or **secured** states and may never be locked.
1	0	Associated memory bank is writeable from the **secured** state but not from the **open** state.
1	1	Associated memory bank is not writeable from any state.
pwd-read/write	**permalock**	**Description**
0	0	Associated password location is readable and writeable from either the **open** or **secured** states.
0	1	Associated password location is permanently readable and writeable from either the **open** or **secured** states and may never be locked.
1	0	Associated password location is readable and writeable from the **secured** state but not from the **open** state.
1	1	Associated password location is not readable or writeable from any state.

2.10.2 Notification Frame

If the Lock command is executed correctly and the tag's response is valid, the Notification Frame is as follows:

Type: 0x01
Command: 0x82
Parameter Length: 0x0010
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Data: 0x00 (Execution successful)
Checksum: 0xE2

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	82	00	10	0E	34	00
EPC（MSB）							
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Parameter	Checksum	End
70	00	E2	7E

If the tag is not within the field or the specified EPC code is incorrect, error code 0x13 will be returned as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x13
Checksum: 0x14

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	13	14	7E

If the Access Password is incorrect, error code 0x16 will be returned, along with the PC+EPC of the tag being operated, as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0x16
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）				Checksum			End	
70				75			7E	

If the operated tag returns an error code defined by the EPC Gen2 protocol, note that only the lower 4 bits of EPC Gen2 error codes are valid. Consequently, the Notification Frame will OR the tag-returned error code with 0xC0 before sending it back.For example, if the address offset or data length specified in the Parameter field is incorrect, or the requested read length exceeds the tag's memory area, the tag will return EPC Gen2 error code 0x04 (Memory Overrun). In this case, the Notification Frame will return error code 0xC4 along with the PC+EPC of the tag being operated, as shown below:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0xC4
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x23

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	C4	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）			Checksum			End		
70			23			7E		

2.11 Kill Kill Tag

2.11.1 Command Frame

This command should be preceded by setting the Select parameter to choose the specific tag for the Kill operation. For a single-tag Kill operation.

Type: 0x00
Command: 0x65
Parameter Length: 0x0004
Kill Password: 0x0000FFFF
Checksum: 0x67

Header	Type	Command	PL（MSB）	PL（LSB）	KP（MSB）		
BB	00	65	00	04	00	00	FF
KP（LSB）		Checksum		End			
70		67		7E			

2.11.2 Notification Frame

If the Kill command is executed correctly and the tag returns a correct CRC, the Notification Frame is as follows:

Type: 0x00
Command: 0x65
Parameter Length: 0x0010
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Data: 0x00 (Execution successful)
Checksum: 0xC5

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	65	00	10	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）			Parameter		Checksum		End	
70			00		C5		7E	

如果该标签没有在场区或者指定的 EPC Code 不对，会返回错误 Code 0x12，如下:

Type: 0x01

Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x12
Checksum: 0x13

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	12	13	7E

If the operated tag returns an error code defined by the EPC Gen2 protocol, the Notification Frame will OR the tag-returned error code with 0xD0 before sending it back.

Note: If the tag has never been assigned a Kill Password—i.e., the Kill Password is all zeros—the tag will not be killed according to the Gen2 protocol. In this case, error code 0xD0 is returned, as shown below:
Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0xD0
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x2F

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	D0	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）				Checksum			End	
70				2F			7E	

2.12 Set Communication Baud Rate

2.12.1 Command Frame

After connecting to the reader, set the subsequent communication baud rate. For example, to set it to 19200, the Command Frame is defined as follows:

Type: 0x00
Command: 0x11
Parameter Length: 0x0002
Pow: 0x00C0 (The baud rate divided by 100, expressed in hexadecimal. For instance, 19200 → 19200/100 = 192 = 0xC0)
Checksum: 0x45

Header	Type	Command	PL（MSB）	PL（LSB）	Pow（MSB）	Pow（LSB）	Checksum	End
BB	00	11	00	02	00	C0	D3	7E

2.12.2 Notification Frame

This command does not have a Notification Frame. After the reader executes the Set Communication Baud Rate command, it will begin communicating with the host at the new baud rate, and the host must reconnect to the reader using this new baud rate.

2.13 Get Query Parameters

2.13.1 Command Frame

Type: 0x00
Command: 0x0D
Parameter Length: 0x0000
Checksum: 0x0D

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	0D	00	00	0D	7E

2.13.2 Notification Frame

Type: 0x01
Command: 0x0D
Parameter Length: 0x0002
Query Parameter: 0x1020
Checksum: 0x40

Header	Type	Command	PL（MSB）	PL（LSB）	Para（MSB）	Para（LSB）	Checksum	End
BB	01	0D	00	02	10	20	40	7E

The parameter is 2 bytes long and is formed by concatenating the specific parameters below bit by bit. The Query parameters corresponding to the above Notification Frame are:
DR = 8, M = 1, TRext = Use pilot tone, Sel = 00, Session = 00, Target = A, Q = 4
Details are as follows:
DR (1 bit): DR = 8 (1'b0), DR = 64/3 (1'b1). Only DR = 8 mode is supported
M (2 bits): M = 1 (2'b00), M = 2 (2'b01), M = 4 (2'b10), M = 8 (2'b11). Only M = 1 mode is supported
TRext (1 bit): No pilot tone (1'b0), Use pilot tone (1'b1). Only the Use pilot tone (1'b1) mode is supported
Sel(2 bit): ALL(2'b00/2'b01), ~SL(2'b10), SL(2'b11)
Session(2 bit): S0(2'b00), S1(2'b01), S2(2'b10), S3(2'b11)
Target(1 bit): A(1'b0), B(1'b1)
Q(4 bit): 4'b0000-4'b1111

2.14 Set Query Parameters

2.14.1 Command Frame

Set the related parameters in the Query command. The parameter is 2 bytes long and is formed by concatenating the following specific parameters bit by bit:
DR (1 bit): DR = 8 (1'b0), DR = 64/3 (1'b1). Only DR = 8 mode is supported
M (2 bits): M = 1 (2'b00), M = 2 (2'b01), M = 4 (2'b10), M = 8 (2'b11). Only M = 1 mode is supported
TRext (1 bit): No pilot tone (1'b0), Use pilot tone (1'b1). Only the Use pilot tone (1'b1) mode is supported
Sel (2 bits): ALL (2'b00/2'b01), ~SL (2'b10), SL (2'b11)
Session (2 bits): S0 (2'b00), S1 (2'b01), S2 (2'b10), S3 (2'b11)
Target (1 bit): A (1'b0), B (1'b1)
Q (4 bits): 4'b0000–4'b1111
If DR = 8, M = 1, TRext = Use pilot tone, Sel = 00, Session = 00, Target = A, Q = 4, the command is as follows:

Type: 0x00
Command: 0x0E
Parameter Length: 0x0002
Query Parameter: 0x1020
Checksum: 0xC6

Header	Type	Command	PL（MSB）	PL（LSB）	Para（MSB）	Para（LSB）	Checksum	End
BB	00	0E	00	02	10	20	40	7E

2.14.2 Notification Frame

If the Set Query Parameters command is executed successfully, the Notification Frame is as follows:

Type: 0x01
Command: 0x0E
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0x10

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	0E	00	01	00	10	7E

2.15 Set Operating Region

2.15.1 Command Frame

Set the reader's operating region. If it is the China 900 MHz band, proceed as follows:

Type: 0x00

Command: 0x07
Parameter Length: 0x0001
Region: 0x01
Checksum: 0x09

Header	Type	Command	PL（MSB）	PL（LSB）	Region	Checksum	End
BB	00	07	00	01	01	09	7E

The region codes for different countries are as follows:

Region	Code
China 900MHz	01
China 800MHz	04
America	02
Europe	03
South Korea	06

2.15.2 Notification Frame

If the Region is configured correctly, the Notification Frame will be:

Type: 0x01
Command: 0x07
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0x09

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	07	00	01	00	09	7E

2.16 Get Operating Region

2.16.1 Command Frame

Type: 0x00
Command: 0x08
Parameter Length: 0x0000
Checksum: 0x08

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	08	00	00	08	7E

2.16.2 Notification Frame

Set the reader's operating Region. If it is the China 900 MHz band, configure it as follows:

Type: 0x01
Command: 0x08
Parameter Length: 0x0001
Region: 0x01
Checksum: 0x0B

Header	Type	Command	PL（MSB）	PL（LSB）	Region	Checksum	End
BB	01	08	00	01	01	0B	7E

The region codes for different countries are as follows:

Region	Code
China 900MHz	01
China 800MHz	04
America	02
Europe	03
South Korea	06

2.17 Set Operating Channel

2.17.1 Command Frame

If using the China 900 MHz band, set the reader's operating channel to 920.375 MHz as follows:

Type: 0x00
Command: 0xAB
Parameter Length: 0x0001
Region: 0x01
Checksum: 0xAD

Header	Type	Command	PL（MSB）	PL（LSB）	Region	Checksum	End
BB	00	AB	00	01	01	AD	7E

China 900 MHz channel parameter calculation formula, where Freq_CH is the channel frequency:

CH_Index = (Freq_CH-920.125M)/0.25M

China 800 MHz channel parameter calculation formula, where Freq_CH is the channel frequency:

CH_Index = (Freq_CH-840.125M)/0.25M

U.S. channel parameter calculation formula, where Freq_CH is the channel frequency:

CH_Index = (Freq_CH-902.25M)/0.5M

Europe channel parameter calculation formula, where Freq_CH is the channel frequency:

CH_Index = (Freq_CH-865.1M)/0.2M

Korea channel parameter calculation formula, where Freq_CH is the channel frequency:

CH_Index = (Freq_CH-917.1M)/0.2M

2.17.2 Notification Frame

If the channel is configured correctly, the Notification Frame will be:

Type: 0x01
Command: 0xAB
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0xAD

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	AB	00	01	00	AD	7E

2.18 Get Operating Channel

2.18.1 Command Frame

In the current reader operating Region, obtain the reader's operating channel as follows:

Type: 0x00
Command: 0xAA
Parameter Length: 0x0000
Checksum: 0xAA

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	AA	00	00	AA	7E

2.18.2 Notification Frame

If the channel retrieval is executed correctly, the Command Frame response will be:

Type: 0x01
Command: 0xAA
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0xAC

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	AA	00	01	00	AC	7E

China 900 MHz channel parameter calculation formula, where Freq_CH is the channel frequency:

Freq_CH = CH_Index * 0.25M + 920.125M

China 800 MHz channel parameter calculation formula, where Freq_CH is the channel frequency:

Freq_CH = CH_Index * 0.25M + 840.125M

U.S. channel parameter calculation formula, where Freq_CH is the channel frequency:

Freq_CH = CH_Index * 0.5M + 902.25M

Europe channel parameter calculation formula, where Freq_CH is the channel frequency:

Freq_CH = CH_Index * 0.2M + 865.1M

Korea channel parameter calculation formula, where Freq_CH is the channel frequency:

Freq_CH = CH_Index * 0.2M + 917.1M

2.19 Set Automatic Frequency Hopping

2.19.1 Command Frame

Enable or disable the automatic frequency-hopping mode. In automatic frequency-hopping mode, if the user has executed the Insert Operating Channel command, the reader randomly selects channels for hopping from the user-defined channel list; otherwise, it randomly selects channels for hopping from the internally preset channel list. The command format is as follows::

Type: 0x00
Command: 0xAD
Parameter Length: 0x0001
Parameter: 0xFF (0xFF sets automatic frequency hopping, 0x00 cancels automatic frequency hopping)
Checksum: 0xAD

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	00	AD	00	01	FF	AD	7E

2.19.2 Notification Frame

If automatic frequency hopping is enabled or disabled successfully, the Notification Frame will be:

Type: 0x00
Command: 0xAD
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0xAF

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	AD	00	01	00	AF	7E

2.20 Insert Operating Channel

2.20.1 Command Frame

The Insert Operating Channel command allows the user to define a custom list of channels for frequency hopping. After this command is executed, the reader will randomly select channels for hopping from the user-defined channel list. The command is defined as follows:

Type: 0x00
Command: 0xA9

Parameter Length: 0x0006
CH Cnt: 0x05
CH List: 0x01 0x02 0x03 0x04 0x05
Checksum: 0xC3

Header	Type	Command	PL（MSB）	PL（LSB）	CH Cnt
BB	00	A9	00	06	05

CH List（MSB）				CH List（LSB）	Checksum	End
01	02	03	04	05	C3	7E

2.20.2 Notification Frame

If executed successfully, the Notification Frame will be:

Type: 0x01
Command: 0xA9
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0xAB

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	A9	00	01	00	AB	7E

2.21 Get Transmit Power

2.21.1 Command Frame

Type: 0x00
Command: 0xB7
Parameter Length: 0x0000
Checksum: 0xB7

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	B7	00	00	B7	7E

2.21.2 Notification Frame

If executed successfully, the Notification Frame will be:

Type: 0x01
Command: 0xB7
Parameter Length: 0x0002
Pow: 0x07D0 (Current power is decimal 2000, i.e., 20 dBm)
Checksum: 0x91

Header	Type	Command	PL（MSB）	PL（LSB）	Pow（MSB）	Pow（LSB）	Checksum	End
BB	01	B7	00	02	07	D0	91	7E

2.22 Set Transmit Power

2.22.1 Command Frame

Type: 0x00
Command: 0xB6
Parameter Length: 0x0002
Pow: 0x07D0 (Current power is decimal 2000, i.e., 20 dBm)
Checksum: 0x8F

Header	Type	Command	PL（MSB）	PL（LSB）	Pow（MSB）	Pow（LSB）	Checksum	End
BB	00	B6	00	02	07	D0	8F	7E

2.22.2 Notification Frame

If the channel retrieval is executed correctly, the Notification Frame will be:

Type: 0x01
Command: 0xB6
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0xB8

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	B6	00	01	00	B8	7E

2.23 Set Continuous Carrier Transmission

2.23.1 Command Frame

Set continuous carrier transmission or disable continuous carrier transmission as follows:

Type: 0x00
Command: 0xB0
Parameter Length: 0x0001
Parameter: 0xFF (0xFF turns on continuous wave, 0x00 turns off continuous wave)
Checksum: 0xB0

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	00	B0	00	01	FF	B0	7E

2.23.2 Notification Frame

If the setting is executed correctly, the Notification Frame will be:

Type: 0x01
Command: 0xB0

Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0xB2

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	B0	00	01	00	B2	7E

2.24 Get Receiver Demodulator Parameters

2.24.1 Command Frame

Obtain the current reader's receiver demodulator parameters. The demodulator parameters include Mixer gain, intermediate frequency amplifier (IF AMP) gain, and signal demodulation threshold. For example:

Type: 0x00
Command: 0xF1
Parameter Length: 0x0000
Checksum: 0xF1

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	F1	00	00	F1	7E

2.24.2 Notification Frame

Type: 0x01
Command: 0xF1
Parameter Length: 0x0004
Mixer gain Mixer_G: 0x03 (Mixer gain is 9 dB)
IF amplifier gain IF_G: 0x06 (IF AMP gain is 36 dB)
Signal demodulation threshold Thrd: 0x01B0 (A lower signal demodulation threshold allows demodulation of tags with lower return RSSI but is less stable; below a certain value, demodulation is impossible. Conversely, a higher threshold requires higher return RSSI from the tag, meaning closer range and greater stability. 0x01B0 is the recommended minimum value.)
Checksum: 0xB0

Header	Type	Command	PL（MSB）	PL（LSB）	Mixer_G	IF_G	Thrd（MSB）	Thrd（LSB）	Checksum	End
BB	01	F1	00	04	03	06	01	B0	B0	7E

Mixer Gain

Type	Mixer_G(dB)
0x00	0
0x01	3
0x02	6
0x03	9
0x04	12
0x05	15
0x06	16

IF AMP Gain

Type	IF_G(dB)
0x00	12
0x01	18
0x02	21
0x03	24
0x04	27
0x05	30
0x06	36
0x07	40

2.25 Set Receiver Demodulator Parameters

2.25.1 Command Frame

Configure the current reader's receiver demodulator parameters. The demodulator parameters include Mixer gain, intermediate frequency amplifier (IF AMP) gain, and signal demodulation threshold. For example:

Type: 0x00
Command: 0xF0
Parameter Length: 0x0004
Mixer gain Mixer_G: 0x03 (Mixer gain is 9 dB)
IF amplifier gain IF_G: 0x06 (IF AMP gain is 36 dB)
Signal demodulation threshold Thrd: 0x01B0 (A lower signal demodulation threshold allows demodulation of tags with lower return RSSI but is less stable; below a certain value, demodulation is impossible. Conversely, a higher threshold requires higher return RSSI from the tag, meaning closer range and greater stability. 0x01B0 is the recommended minimum value.)
Checksum: 0xAE

Header	Type	Command	PL（MSB）	PL（LSB）	Mixer_G	IF_G	Thrd（MSB）	Thrd（LSB）	Checksum	End
BB	00	F0	00	04	03	06	01	B0	AE	7E

Mixer Gain

Type	Mixer_G(dB)
0x00	0
0x01	3
0x02	6
0x03	9
0x04	12
0x05	15
0x06	16

IF AMP Gain

Type	IF_G(dB)
0x00	12
0x01	18
0x02	21
0x03	24
0x04	27
0x05	30
0x06	36
0x07	40

2.25.2 Notification Frame

If the channel retrieval is executed correctly, the Notification Frame will be:

Type: 0x01
Command: 0xF0
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0xF2

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	F0	00	01	00	F2	7E

2.26 Test the RF Input Blocking Signal

2.26.1 Command Frame

Test the RF input blocking signal (Scan Jammer), used to detect the blocking signal level on each channel of the reader antenna in the current Region. For example:

Type: 0x00
Command: 0xF2
Parameter Length: 0x0000
Checksum: 0xF2

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	F2	00	00	F2	7E

2.26.2 Notification Frame

If operating under the China 900 MHz band with a total of 20 channels, and the Scan Jammer test of the RF input blocking signal is executed successfully, the Notification Frame will be:

Type: 0x01
Command: 0xF2
Parameter Length: 0x0016
CH_L: 0x00 (Start test channel index is 0)
CH_H: 0x13 (End test channel index is 19)

Channel blocking signal JMR: 0xF2F1F0EFECEAE8EAECEEF0F1F5F5F5F6F5F5F5F5 (Each channel's blocking signal JMR is represented by a signed byte; for example, 0xF2 equals -14 dBm)

Checksum: 0xDD

Header	Type	Command	PL（MSB）	PL（LSB）	CH_L	CH_H
BB	01	F2	00	16	00	13

JMR（MSB）																		
F2	F1	F0	EF	EC	EA	E8	EA	EC	EE	F0	F1	F5	F5	F5	F6	F5	F5	F5

JMR（LSB）	Checksum	End
F5	DD	7E

2.27 Test Channel RSSI

2.27.1 Command Frame

Test the RF input RSSI signal level to determine whether any reader is operating in the current environment. For example:

Type: 0x00

Command: 0xF3

Parameter Length: 0x0000

Checksum: 0xF3

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	F3	00	00	F3	7E

2.27.2 Notification Frame

If operating under the China 900 MHz band with a total of 20 channels, and the RSSI detection for each channel is executed successfully, the Notification Frame will be:

Type: 0x01

Command: 0xF3

Parameter Length: 0x0016

CH_L: 0x00 (Start test channel index is 0)

CH_H: 0x13 (End test channel index is 19)

Channel blocking signal JMR: 0xF2F1F0EFECEAE8EAECEEF0F1F5F5F5F6F5F5F5F5 (Each channel's blocking signal JMR is represented by a signed byte; for example, 0xF2 equals -14 dBm)

Checksum: 0xA5

Header	Type	Command	PL（MSB）	PL（LSB）	CH_L	CH_H
BB	01	F3	00	16	00	13

RSSI（MSB）																		
BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA

RSSI（LSB）	Checksum	End
BA	A5	7E

2.28 Control IO Port

2.28.1 Command Frame

Set the direction of the IO ports, read the IO level, and set the IO level as follows:

Type: 0x00
Command: 0x1A
Parameter Length: 0x0003
Parameter: 0x00 0x04 0x01
Checksum: 0x22

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter			Checksum	End
BB	01	1A	00	03	00	04	01	22	7E

Parameter Description:

Num	Description	Size	Introduction
0	Parameter0	1 Byte	Operation Type Selection: 0x00: Set IO direction 0x01: Set IO level 0x02: Read IO level; the target pin is specified in Parameter 1.
1	Parameter1	1 Byte	Parameter value range: 0x01~0x04, corresponding to the target ports IO1~IO4 respectively
2	Parameter2	1 Byte	Parameter value: 0x00 or 0x01 Parameter0 / Parameter2 / Description: 0x00 / 0x00 / IO configured as input mode 0x00 / 0x01 / IO configured as output mode 0x01 / 0x00 / Set IO output to low level 0x01 / 0x01 / Set IO output to high level When Parameter 0 is 0x02, this parameter has no effect.

2.28.2 Notification Frame

Type: 0x01
Command: 0x1A
Parameter Length: 0x0003
Parameter: 0x00 0x04 0x01
Checksum: 0x23

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter			Checksum	End
BB	01	1A	00	03	00	04	01	23	7E

参数说明:

Num	Description	Size	说明
0	Parameter0	1 Byte	Operation Type Selection: 0x00: Set IO direction 0x01: Set IO level 0x02: Read IO level; the target pin is specified in Parameter 1.

<table>
<tr><td>1</td><td>Parameter1</td><td>1 Byte</td><td>Parameter value range: 0x01~0x04, corresponding to the target ports IO1~IO4 respectively</td></tr>
<tr><td>2</td><td>Parameter2</td><td>1 Byte</td><td>Parameter value: 0x00 or 0x01
<table>
<tr><th>Parameter0</th><th>Parameter2</th><th>Description</th></tr>
<tr><td>0x00</td><td>0x00</td><td>Indicates IO configuration failure</td></tr>
<tr><td>0x00</td><td>0x01</td><td>Indicates IO configuration success</td></tr>
<tr><td>0x01</td><td>0x00</td><td>Indicates setting IO output failed</td></tr>
<tr><td>0x01</td><td>0x01</td><td>Indicates setting IO output succeeded</td></tr>
<tr><td>0x02</td><td>0x00</td><td>Indicates the corresponding port is at low level</td></tr>
<tr><td>0x02</td><td>0x01</td><td>Indicates the corresponding port is at high level</td></tr>
</table></td></tr>
</table>

2.29 Module sleep

2.29.1 Command Frame

The module sleep command allows the module to remain in a low-power sleep mode. After the module enters sleep, sending any byte through the serial port will wake it up, but that byte will be discarded. The first command received after the module has slept will produce no response, because the first character of that command is discarded. This command causes the M100/QM100 chip to perform a power-down reset. After waking up, the module immediately re-downloads firmware to the M100/QM100 chip and re-applies certain parameters to the module (including the power, frequency, frequency-hopping mode, sleep duration, and receive demodulator parameters configured before sleep, but excluding Select mode and Select parameters). Therefore, some parameters may need to be reconfigured. The command is as follows:

Type: 0x00
Command: 0x17
Parameter Length: 0x0000
Checksum: 0x17

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	17	00	00	17	7E

2.29.2 Notification Frame

If executed successfully, the Notification Frame is:

Type: 0x01
Command: 0x17
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0x19

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	17	00	01	00	19	7E

2.30 Module idle sleep time

2.30.1 Command Frame

This command sets the duration of inactivity after which the module automatically enters sleep mode. Once asleep, the module can be awakened by sending any character through the serial port, but that character will be discarded. The first command received after the module has slept will have no response because its first character is discarded. This command resets the M100/QM100 chip; after waking, the module immediately re-downloads firmware to the M100/QM100 chip and re-applies certain parameters to the module (including the power, frequency, frequency-hopping mode, sleep duration, and receive demodulator parameters configured before sleep, but excluding Select mode and Select parameters). Therefore, some parameters may need to be reconfigured. The command is as follows:

Type: 0x00
Command: 0x1D
Parameter Length: 0x0001
Parameter: 0x02 (Sleeps after 2 minutes of inactivity; range 1–30 minutes, 0x00 means no auto-sleep)
Checksum: 0x17

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	00	1D	00	01	02	20	7E

2.30.2 Notification Frame

If executed successfully, the Notification Frame is:

Type: 0x01
Command: 0x1D
Parameter Length: 0x0001
Parameter: 0x02
Checksum: 0x21

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	1D	00	01	02	21	7E

2.31 IDLE Mode

2.31.1 Command Frame

This command allows the module to enter IDLE mode. In IDLE mode, all analog and RF power supplies are turned off—except for the digital section and communication interface—to reduce power consumption when the module is not active. After the module enters IDLE mode, communication with the host remains normal, and all previously set parameters are retained; the module can still respond to commands from the host. The first inventory operation (or any read/write command that requires interaction with a tag)

performed after entering IDLE mode will bring the module back to normal operation, but its success rate may be lower because the RF circuitry is still stabilizing. Subsequent inventory and other operations will return to normal performance. The command is as follows:

Type: 0x00
Command: 0x04
Parameter Length: 0x0003
IDLE Mode Enter: 0x01 (Enter IDLE mode; 0x00: exit IDLE mode)
Reserved: 0x01 (Reserved, fixed at 0x01)
IDLE Time: 0x03 ((Automatically enters IDLE mode after 3 minutes of inactivity; range 0–30 minutes, 0x00 means no automatic entry into IDLE mode)
Checksum: 0x0C

Header	Type	Command	PL（MSB）	PL（LSB）	Enter	Reserved	IDLE Time	Checksum	End
BB	01	04	00	03	01	01	03	0C	7E

2.31.2 Notification Frame

If executed successfully, the Notification Frame is:

Type: 0x01
Command: 0x04
Parameter Length: 0x0001
Parameter: 0x00 (Executed successfully)
Checksum: 0x06

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	04	00	01	00	06	7E

2.32 NXP ReadProtect/Reset ReadProtect Command

2.32.1 Command Frame

NXP G2X tags support the ReadProtect/Reset ReadProtect commands. When the ReadProtect command is executed successfully, the tag's ProtectEPC and ProtectTID bits are set to '1', placing the tag in a data-protected state. To return the tag from the data-protected state to normal operation, the Reset ReadProtect command must be issued. Before sending this command, set the Select parameters to target the specific tag for the operation.

Type: 0x00
Command: 0xE1
Parameter Length: 0x0005
Access Password: 0x0000FFFF
ReadProtect/Reset ReadProtect: 0x00 (0x00 means execute ReadProtect, 0x01 means execute Reset ReadProtect)
Checksum: 0x0B

Header	Type	Command	PL（MSB）	PL（LSB）	AP（MSB）			AP（LSB）	Reset	Checksum	End
BB	00	E1	00	05	00	00	FF	FF	00	E4	7E

2.32.2 Notification Frame

If the ReadProtect command executes correctly, the Notification Frame is:

Type: 0x01
Command: 0xE1
Parameter Length: 0x0010
URL Length: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Parameter: 0x00 (Execution successful)
Checksum: 0x3D

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E1	00	10	0E	30	00
EPC（MSB）							
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）			Parameter		Checksum		End
70			00		3D		7E

If the Reset ReadProtect command executes correctly, the Notification Frame is:

Type: 0x01
Command: 0xE2
Parameter Length: 0x0010
URL Length: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Parameter: 0x00 (Execution successful)
Checksum: 0x3E

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E2	00	10	0E	30	00
EPC（MSB）							
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）			Parameter		Checksum		End
70			00		3E		7E

If, during the execution of the ReadProtect command (Set/Reset parameter set to 0x00), the tag is not within the field, the specified EPC code is incorrect, or the tag fails to respond, error code 0x2A will be returned as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x2A
Checksum: 0x2B

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	2A	2B	7E

If, during the execution of the Reset ReadProtect command (Set/Reset parameter set to 0x01), the tag is not within the field or the specified EPC code is incorrect, error code 0x2B will be returned as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x2B
Checksum: 0x2C

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	2B	2C	7E

If the Access Password is incorrect, error code 0x16 will be returned, along with the PC+EPC of the tag being operated, as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0x16
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）				Checksum			End	
70				75			7E	

2.33 NXP Change EAS Command

2.33.1 Command Frame

NXP G2X tags support the Change EAS command. When the Change EAS command is executed successfully, the tag's PSF bit is set to '1' or '0' accordingly. When the PSF bit is '1', the tag will respond to the EAS_Alarm command; otherwise, it will not respond to the EAS_Alarm command. Before issuing this command, set the Select parameters to target the desired tag.

The Change EAS command frame is defined as follows:

Type: 0x00
Command: 0xE3

Parameter Length: 0x0005
Access Password: 0x0000FFFF
Set/Reset: 0x0002
Checksum: 0x45

Header	Type	Command	PL（MSB）	PL（LSB）	AP（MSB）			AP（LSB）	PSF	Checksum	End
BB	00	E3	00	05	00	00	FF	FF	01	E7	7E

2.33.2 Notification Frame

If the Change EAS command executes correctly, the Notification Frame is:

Type: 0x01
Command: 0xE3
Parameter Length: 0x0010
URL Length: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Data: 0x00 (Execution successful)
Checksum: 0x3F

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E3	00	10	0E	30	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Parameter	Checksum	End
70	00	3F	7E

If, during the execution of the Change EAS command, the tag is not within the field, the specified EPC code is incorrect, or the tag fails to respond, error code 0x1B will be returned as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x1B
Checksum: 0x1C

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	1B	1C	7E

If the Access Password is incorrect, error code 0x16 will be returned, along with the PC+EPC of the tag being operated, as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0x16
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70

Checksum: 0x75

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Checksum	End
70	75	7E

2.34 NXP EAS_Alarm Command

2.34.1 Command Frame

NXP G2X tags support the EAS_Alarm command. When a tag receives the EAS_Alarm command, it immediately returns a 64-bit EAS-Alarm code. Note that the tag responds to the EAS_Alarm command only when its PSF bit is set to '1'; otherwise, it will not respond. This command is suitable for electronic article surveillance (EAS) systems.

Type: 0x00
Command: 0xE4
Parameter Length: 0x0000
Checksum: 0xE4

Header	Type	Command	PL（MSB）	PL（LSB）	Checksum	End
BB	00	E4	00	00	E4	7E

2.34.2 Notification Frame

If the EAS_Alarm command executes successfully and a tag responds with the correct 64-bit EAS-Alarm code, the Notification Frame is:

Type: 0x01
Command: 0xE4
Parameter Length: 0x0001
Parameter: 0x00
Checksum: 0x80

Header	Type	Command	PL（MSB）	PL（LSB）
BB	01	E4	00	08

EAS-Alarm code（MSB）							EAS-Alarm code（LSB）	Checksum	End
69	0A	EC	7C	D2	15	D8	F9	80	7E

If, during the execution of the EAS_Alarm command, no tag responds, error code 0x1D will be returned as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x1D

Checksum: 0x1E

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	1D	1E	7E

2.35 NXP ChangeConfig Command

2.35.1 Command Frame

Certain NXP G2X tag series (such as the G2iM and G2iM+) support the ChangeConfig command, which can be used to read or modify the 16-bit Config-Word of an NXP G2X tag. The Config-Word is located in memory Bank 01 (the EPC bank) at address 20h (word address) and can be read with a standard Read command. When the tag is in the Secured state, the Config-Word can be rewritten; note that rewriting flips the corresponding bits of the Config-Word—writing a '1' inverts the targeted bit ('1' becomes '0', and '0' becomes '1'), whereas writing a '0' leaves the bit unchanged. Before issuing this command, set the Select parameters to target the specific tag.

Type: 0x00
Command: 0xE0
Parameter Length: 0x0006
Access Password: 0x0000FFFF
Config-Word: 0x0000 When all zeros, the tag returns the unchanged Config-Word, equivalent to a read.
Checksum: 0xE4

Header	Type	Command	PL（MSB）	PL（LSB）	AP（MSB）		
BB	00	E0	00	06	00	00	FF

AP（LSB）	Config（MSB）	Config（LSB）	Checksum	End
FF	00	00	E4	7E

2.35.2 Notification Frame

If the ChangeConfig command executes correctly, the Notification Frame is:

Type: 0x01
Command: 0xE0
Parameter Length: 0x0011
URL Length: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Config-Word: 0x0041
Checksum: 0x7E

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E0	00	11	0E	30	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Config（MSB）	Config（LSB）	Checksum	End
70	00	41	7E	7E

If, during the execution of the ChangeConfig command, the tag is not within the field, the specified EPC code is incorrect, or the tag fails to respond, error code 0x1A will be returned as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x1A
Checksum: 0x1B

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	1A	1B	7E

If the Access Password is incorrect, error code 0x16 will be returned, along with the PC+EPC of the tag being operated, as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0x16
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）				Checksum			End	
70				75			7E	

2.36 Impinj Monza QT Command

2.36.1 Command Frame

Impinj Monza 4 QT tags support the QT command, which can modify the tag’s QT Control word. Setting the QT_SR bit shortens the operating range when the tag is in, or about to enter, either the Open (public) or Secured (private) state. Changing the QT_MEM bit switches the tag between the Public Memory Map and the Private Memory Map. Before issuing this command, set the Select parameters to target the desired tag.

The QT command frame is defined as follows; in this example, the QT_MEM bit is set to 1 and written to the tag’s non-volatile memory:

Type: 0x00
Command: 0xE5
Parameter Length: 0x0008
Access Password: 0x0000FFFF
Read/Write: 0x01 (0x00: Read，0x01: Write)

Persistence: 0x01 (0x00: write to the tag's volatile memory, 0x01: write to the tag's non-volatile memory)
Payload: 0x4000 (QT Control, the two most significant bits are QT_SR and QT_MEM respectively)
Checksum: 0x2D

Header	Type	Command	PL（MSB）	PL（LSB）	AP（MSB）		
BB	00	E5	00	08	00	00	FF

AP（LSB）	Read/Write	Persistence	Payload0	Payload1	Checksum	End
FF	01	01	40	00	2D	7E

2.36.2 Notification Frame

If the QT command executes correctly and the Read/Write data field is 0x00, the Notification Frame is:

Type: 0x01
Command: 0xE5
Parameter Length: 0x0011
URL Length: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
QT Control Word: 0x0000
Checksum: 0x42

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E5	00	11	0E	30	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	QT Control0	QT Control1	Checksum	End
70	00	00	42	7E

If the QT command executes correctly and the Read/Write data field is 0x01, the Notification Frame is:

Type: 0x01
Command: 0xE6
Parameter Length: 0x0010
URL Length: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Parameter: 0x00
Checksum: 0x42

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E6	00	10	0E	30	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Parameter	Checksum	End
70	00	42	7E

If, during the execution of the QT command, the tag is not within range, the specified EPC code is incorrect, or the tag fails to respond, error code 0x2E will be returned as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x2E
Checksum: 0x2F

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	2E	2F	7E

If the Access Password is incorrect, error code 0x16 will be returned, along with the PC+EPC of the tag being operated, as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0x16
URL Length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）				Checksum			End	
70				75			7E	

2.37 BlockPermalock Command

2.37.1 Command Frame

The BlockPermalock command can permanently lock specific blocks in the user memory area or read the lock status of those blocks. Before issuing this command, set the Select parameters to target the desired tag.The BlockPermalock command frame is defined as follows; in this example, the BlockPermalock status is written to permanently lock Blocks 5, 6, and 7:

Type: 0x00
Command: 0xD3
Parameter Length: 0x0009
Access Password: 0x0000FFFF
Read/Lock: 0x00 (0x00: Read，0x01: Lock)
BlockPtr: 0x0000 (Start block address of the mask, in units of 16 blocks)

BlockRange: 0x01 (in units of 16 blocks)

Mask: 0x0700 (When the Read/Lock data field is 0x00, i.e., read status, this data field is omitted)

Checksum: 0xE8

Header	Type	Command	PL（MSB）	PL（LSB）	AP（MSB）			AP（LSB）
BB	00	D3	00	0B	00	00	FF	FF

Read/Lock	MemBank	BlockPtr1	BlockPtr0	BlockRange	Mask（MSB）	Mask（LSB）	Checksum	End
01	03	00	00	01	07	00	E8	7E

2.37.2 Notification Frame

If the BlockPermalock command executes correctly and the Read/Lock data field is 0x00, the Notification Frame is:

Type: 0x01

Command: 0xD3

Parameter Length: 0x0012

URL Length: 0x0E

PC: 0x3000

EPC: 0xE20030166606006911609F94

BlockRange: 0x01

Data: 0x0700

Checksum: 0xCD

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	D3	00	12	0E	30	00

EPC（MSB）										
E2	00	30	16	66	06	00	69	11	60	9F

EPC（LSB）	BlockRange	Data（MSB）	Data（LSB）	Checksum	End
70	01	07	00	CD	7E

If the BlockPermalock command executes correctly and the Read/Lock data field is 0x01, the Notification Frame is:

Type: 0x01

Command: 0xD4

Parameter Length: 0x0010

URL Length: 0x0E

PC: 0x3000

EPC: 0xE20030166606006911609F94

Parameter: 0x00 (Execution successful)

Checksum: 0xC4

Header	Type	Command	PL（MSB）	PL（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	D4	00	10	0E	30	00

EPC（MSB）										
E2	00	30	16	66	06	00	69	11	60	9F

EPC（LSB）	Parameter	Checksum	End
94	00	C4	7E

If, during the execution of the BlockPermalock command, the tag is not within the field, the specified EPC code is incorrect, or the tag fails to respond, error code 0x14 will be returned as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0001
Parameter: 0x14
Checksum: 0x15

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	14	15	7E

If, during the execution of the BlockPermalock command, the tag returns an EPC Gen2 protocol error code (error-code), the Notification Frame will OR the tag’s returned error code with 0xE0 before reporting it, because only the lower four bits of EPC Gen2 error codes are valid. For example, if the Parameter BlockPtr exceeds the tag’s memory block range, the tag will return error code 0x03 (Memory Overrun). Consequently, the Notification Frame will return error code 0xE3 along with the PC + EPC of the tag being operated on, as shown below:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0010
Parameter: 0xA3
URL Length: 0x0E
PC: 0x3000
EPC: 0xE20030166606006911609F94
Checksum: 0xD2

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	E3	0E	30	00
EPC（MSB）								
E2	00	30	16	66	06	00	69	11	60	9F
EPC（LSB）				Checksum			End	
94				D2			7E	

If the Access Password is incorrect, error code 0x16 will be returned, along with the PC+EPC of the tag being operated, as follows:

Type: 0x01
Command: 0xFF
Parameter Length: 0x0016
Parameter: 0x16
URL Length: 0x0E
PC: 0x3000
EPC: 0xE20030166606006911609F94

Checksum: 0x05

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00
EPC（MSB）								
E2	00	30	16	66	06	00	69	11	60	9F
EPC（LSB）			Checksum		End			
94			05		7E			

3、Command Summary

Code	Description
0x03	Get Reader Module Information
0x22	Single Polling Command
0x27	Multiple Polling Command
0x28	Stop Multiple Polling Command
0x0C	Set Select Parameter Command
0x0B	Get Select Parameter
0x12	Set to send the Select command
0x39	Read Tag Memory Area
0x49	Write Tag Memory Area
0x82	Lock Tag Memory Area
0x65	Kill KillTag
0x0D	Get Query Parameter
0x0E	Set Query Parameter
0x07	Set Operating Region
0xAB	Set Operating Channel
0xAA	Get Operating Channel
0xAD	Set Automatic Frequency Hopping
0xB7	Get Transmit Power
0xB6	Set Transmit Power
0xB0	Set Continuous Carrier Transmission
0xF1	Get Receiver Demodulator Parameters
0xF0	Set Receiver Demodulator Parameters
0xF2	Test the RF input blocking signal
0xF3	Test Channel RSSI
0x1A	Control IO Port
0x17	Module sleep
0x1D	Set Module idle sleep time
0xE0	NXP ChangeConfig Command
0xE1	NXP ReadProtec/Reset ReadProtect Command
0xE3	NXP Change EAS Command
0xE4	NXP EAS-Alarm Command
0xE5/0xE6	Impinj Monza 4 QT Command
0xD3/0xD4	BlockPermalock Command

4 、 Summary of Notification Frames for Command Frame Execution Failures

If the Command Frame fails to execute, the M100 chip sends a failure Notification Frame to the host. All failure Notification Frames use Command 0xFF. If the tag's EPC has not been obtained before the failure, the Parameter field contains a fixed 1-byte error code. If the tag's EPC has been obtained before the failure, the Notification Frame contains a 1-byte error code followed by the tag's PC + EPC data.

For example, if a polling Command Frame fails due to no tag response or a Data CRC check error, error code 0x15 is returned as follows:

Type: 0x01

Command: 0xFF (0xFF indicates Command Frame execution failure)

Parameter Length: 0x01

Parameter: 0x15 (The error codes are summarized as follows)

Checksum: 0x16

Header	Type	Command	PL（MSB）	PL（LSB）	Parameter	Checksum	End
BB	01	FF	00	01	15	16	7E

The error codes are summarized as follows:

Type	Code	Description
Command Error	0x17	Command error in Command Frame
FHSS Fail	0x20	Frequency-hopping channel search timed out. All channels were occupied during this period.
Inventory Fail	0x15	Polling operation failed. No tag returned or Data CRC check error.
Access Fail	0x16	Failed to access the tag, possibly due to an incorrect access password.
Read Fail	0x09	Failed to read the tag’s data memory area. The tag did not respond or a Data CRC check error occurred.
Read Error	0xA0 | Error code	Read Tag Memory Area error. The returned code is obtained by OR-ing 0xA0 with the Error Code. Refer to the table below for error code details.
Write Fail	0x10	Failed to write the tag's data memory area. The tag did not respond or a Data CRC check error occurred.
Write Error	0xB0 | Error code	Write Tag Memory Area error. The returned code is obtained by OR-ing 0xB0 with the Error Code. Refer to the table below for detailed error code information.
Lock Fail	0x13	Failed to lock the tag's data memory area. The tag did not respond or a Data CRC check error occurred.
Lock Error	0xC0 | Error code	Lock Tag Memory Area error. The returned code is obtained by OR-ing 0xC0 with the Error Code. Refer to the table below for detailed error code information.
Kill Fail	0x12	Failed to kill the tag; the tag did not respond or a Data CRC check error occurred.
Kill Error	0xD0 | Error code	Kill Tag error. The returned code is obtained by OR-ing 0xC0 with the Error Code. For error code details, refer to the tag error codes defined in the EPC Gen2 protocol.

Type	Code	Description
BlockPermalock Fail	0x14	BlockPermalock execution failed. The tag did not respond or a Data CRC check error occurred.
BlockPermalock Error	0xE0 | Error code	BlockPermalock error. The returned code is obtained by OR-ing 0xE0 with the Error Code. For error code details, refer to the tag error codes defined in the EPC Gen2 protocol.

NXP G2X tag-specific command error codes:

Type	Code	Description
ChangeConfig Fail	0x1A	ChangeConfig command failed. The tag did not return data or a Data CRC check error occurred.
ReadProtect Fail	0x2A	ReadProtect command failed. The tag returned no data or a Data CRC check error occurred.
Reset ReadProtect Fail	0x2B	Reset ReadProtect command failed. The tag returned no data or a Data CRC check error occurred.
Change EAS Fail	0x1B	Change EAS command failed. The tag returned no data or a Data CRC check error occurred.
EAS_Alarm Fail	0x1D	EAS_Alarm command failed. No tag returned the correct Alarm Code.
Error codes returned by the tag for proprietary commands	0xE0 | Error code	Error codes returned by proprietary commands; the error code is obtained by OR-ing 0xE0 with the error code returned by the tag.

Impinj Monza QT tag-specific command error codes:

Type	Code	Description
QT Fail	0x2E	QT command failed. The tag returned no data or a Data CRC check error occurred.
Error codes returned by the tag for proprietary commands	0xE0 | Error code	Error codes returned by proprietary commands; the error code is obtained by OR-ing 0xE0 with the error code returned by the tag.

Error codes returned by tags in the EPC Gen2 protocol:

Tag error-code

Error-code Support	Error Code	Error Code Name	Error Description
Error-specific	00000000_2	Other error	All other errors not specified in this table.
	00000011_2	Memory overrun	The specified tag memory area does not exist, or the tag does not support the specified EPC length, such as XPC.
	00000100_2	Memory locked	The specified tag memory area is locked and/or permanently locked, and its lock state is non-writable or non-readable.
	00001011_2	Insufficient power	The tag did not receive sufficient power to perform the write operation.
Non-specific	00001111_2	Non-specific error	The tag does not support returning an error code.