H-ZONE

Communication Protocol
V2.1

版本记录

Version	Day	Description
V1.1	2013/10/15	Initial version.
V1.2	2013/12/19	Add the IO control command
V2.0	2014/5/20	Modify Read, Write, Lock, Kill commands by removing PC and EPC data field
V2.1	2015/04/28	Add NXP and Impinj Monza tags' commands.

1. Contents

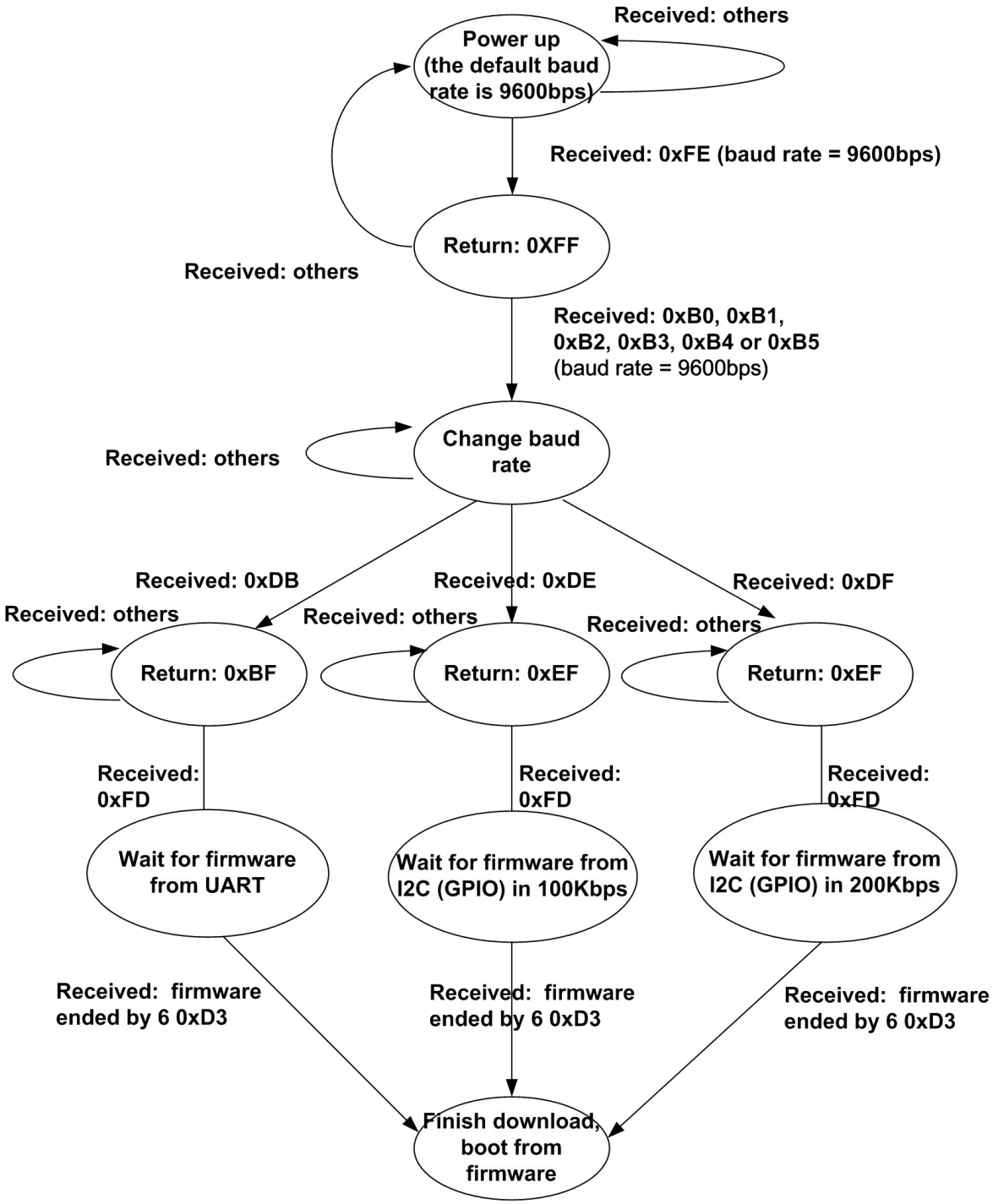
2. Internal MCU

M100 has an internal 8bit 8051 MCU, which has 256Byte RAM, 16Kbyte program ROM and 3 timer (In the firmware, the timer2 is used for baud rate control, the timer0 is used for frequency hopping control and the timer1 is reserved for user). And there is 8Kbyte data RAM which can be used by the 8051 MCU and the MODEM. In the receiving phase, the data RAM can NOT be accessed by MCU.

There is no internal flash in M100. The firmware of MCU could be download by UART or I2C (GPIO emulated: P1.0, P1.1) interface.

UART interface frame has 8bit data and 1bit stop, no parity.

The handshake protocol for firmware download at power up or reset is as below:



Baud Rate Settings

Parameter	Baud Rate(bps)
0xB0	9600
0xB1	19200
0xB2	28800
0xB3	38400
0xB4	57600
0xB5	115200

3. Command frame

3.1. Command Frame structure

Every command includs: Header, Command Type, Command Code, Parameter Length (PL), Checksum, End.

Example:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	00	07	00	01	01	09	DD

Header: 0xAA
Type: 0x00
Command: 0x07
PL: 0x0001
Parameter: 0x01
Checksum: 0x09
End: 0xDD

Checksum is calculated by: Sum from Type to Parameter. And take the least significant Byte of the sum.

3.2. Command Type

Type	Description
0x00	Command frame: command received by firmware
0x01	Response frame: firmware response to the received command
0x02	Notice frame: firmware initiated report

Every received command has an corresponding response which returns the command execution result.

Notice frame is used for single or multiple inventory. Firmware sends notice frame every time when read tag successfully.

4. Command Definition

4.1 Get Module Information

4.1.1. Command frame

This command can get the information of modules such as hardware version, software version and manufacturer information.

Frame type : 0x00

Command code : 0x03

Parameter :

Hardware version : 0x00

Software version : 0x01

Manufacturer: 0x02

For example, get the hardware version:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	00	03	00	01	00	04	DD

Type: 0x00

Command: 0x03

PL: 0x0001

Parameter: 0x00(hardware version)

Checksum: 0x04

4.1.2. Response frame

Frame type : 0x01

Command code : 0x03

Data : variables (ASCII codes)

The response data first byte is the information type, followed by the detailed message in ASCII codes.

The response of getting hardware version is as blow :

Header	Type	Command	PL(MSB)	PL(LSB)	Info Type	Info	
AA	01	03	00	10	00	4D (‘M’)	31 (‘1’)
30 (‘0’)	30 (‘0’)	20 (‘ ‘)	32 (‘2’)	36 (‘6’)	64 (‘d’)	42 (‘B’)	6D (‘m’)
					Checksum	**End**	
20 (‘ ‘)	56 (‘V’)	31(‘1’)	2E (‘.’)	30 (‘0’)	22	DD	

Type: 0x01

Command: 0x03

PL: 0x0010

Info Type: 0x00

Info: 4D 31 30 30 20 32 36 64 42 6D 20 56 31 2E 30

Checksum: 0x22

4.2. Single Inventory

4.2.1. Command frame

When received this command, the reader chip will do the inventory once. The power amplifier will be opened at the beginning of the inventory and be closed after the inventory. Select parameter settings please refer to section 4.4. Query parameter settings please refer to section 4.11.

Single inventory command is as below:

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	22	00	00	22	DD

Type: 0x00
Command: 0x22
PL: 0x0000
Checksum: 0x22

4.2.2. Notice frame

After the firmware received a single inventory command, and successfully read a tag (CRC check passed), a notice frame including RSSI, PC, EPC and CRC will be sent. If multiple tags are read in one inventory, notice frames of the same number of successfully read tag will be sent.

The notice frame is as below:

Header	Type	Command	PL(MSB)	PL(LSB)	RSSI	PC(MSB)	PC(LSB)
AA	02	22	00	11	C9	34	00
EPC(MSB)							
30	75	1F	EB	70	5C	59	04
			EPC(LSB)	**CRC(MSB)**	**CRC(LSB)**	**Checksum**	**End**
E3	D5	0D	70	3A	76	EF	DD

Type: 0x02
Command: 0x22
PL: 0x0011
RSSI: 0xC9
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
CRC: 0x3A76
Checksum: 0xEF

RSSI is a received signal strength indicator at the RF input port of the reader chip. It's a complementally coded signed hex in the unit of dBm. E.g. an RSSI of 0xC9 means the tag returned signal power at the input port of the reader chip is -55dBm。

4.2.3. Response frame

If there is no tag response, or a CRC error occurred, an error code of 0x15 will be sent in response frame：

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	FF	00	01	15	16	DD

Type: 0x01
Command: 0xFF
PL: 0x01

Parameter: 0x15

Checksum: 0x16

4.3. Multiple Inventory

4.3.1. Command frame

When received this command, the reader chip will do the inventory by multiple times. The value of inventory time could be between 0 to 65535.

The following is an example for 10000 times inventory command:

Header	Type	Command	PL(MSB)	PL(LSB)	Reserved	CNT(MSB)	CNT(LSB)
AA	00	27	00	03	22	27	10
Checksum	**End**						
83	DD						

Type: 0x00
Command: 0x27
PL: 0x0003
Reserved: 0x22
CNT: 0x2710
Checksum: 0x83

4.3.2. Notice frame

The notice frame of multiple inventory is the same as single inventory:

Header	Type	Command	PL(MSB)	PL(LSB)	RSSI	PC(MSB)	PC(LSB)
AA	02	22	00	11	C9	34	00
EPC(MSB)							
30	75	1F	EB	70	5C	59	04
			EPC(LSB)	**CRC(MSB)**	**CRC(LSB)**	**Checksum**	**End**
E3	D5	0D	70	3A	76	EF	DD

Type: 0x02
Command: 0x22
PL: 0x0011
RSSI: 0xC9
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
CRC: 0x3A76
Checksum: 0xEF

4.3.3. Response frame

If there is no tag response, or a CRC error occurred, an error code of 0x15 will be sent in response frame:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	FF	00	01	15	16	DD

Type: 0x01
Command: 0xFF
PL: 0x01
Parameter: 0x15
Checksum: 0x16

4.4. Stop Multiple Inventory

4.4.1. Command frame

The multiple inventory could be stopped (not pause) immediately by this command:

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	28	00	00	28	DD

Type: 0x00
Command: 0x28
PL: 0x0000
Checksum: 0x28

4.4.2. Response frame

If the multiple inventory is stopped successfully, a response frame will be return:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	28	00	01	00	2A	DD

Type: 0x01
Command: 0x28
PL: 0x0001
Parameter: 0x00
Checksum: 0x2A

4.5. Set Select Parameters

4.5.1. Command frame

This command will set select parameters and set Select Mode to 0x02. And the select will be send before inventory by default. Example (select tags which has the same code from bit 0x20 to bit 0x80):

Header	Type	Command	PL(MSB)	PL(LSB)	SelParam	Ptr(MSB)	
AA	00	0C	00	13	01	00	00
	Ptr(LSB)	**MaskLen**	**Truncate**	**Mask(MSB)**			
00	20	60	00	30	75	1F	EB
							Mask(LSB)
70	5C	59	04	E3	D5	0D	70
Checksum	**End**						
AD	DD						

Type: 0x00
Command: 0x0C
PL 0x0013
SelParam: 0x01 (Target: 3'b000, Action: 3'b000, MemBank: 2'b01)
Ptr: 0x00000020(in bit) from EPC first bit
MaskLen: 0x60
Truncate: 0x00(0x00: Disable truncation，0x80: Enable truncation)
Mask: 0x30751FEB705C5904E3D50D70
Checksum: 0xAD

SelParam is a one byte parameter which contains 3-bits Target, 3-bits Action and 2-bits MemBank.
The meaning of MemBank is:
2'b00: RFU Memory Bank
2'b01: EPC Memory Bank
2'b02: TID Memory Bank
2'b03: User Memory Bank
The meanings of Target and Action please refer to EPC UHF Class 1 Gen 2 Standard Protocol.

4.5.2. Response frame

When select parameters are set successfully, a response frame will be return:

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

Type: 0x01
Command: 0x0C
PL: 0x0001
Data: 0x00
Checksum: 0x0E

4.6. Set Select Mode

4.6.1. Command frame

After set select parameters, the select command could be set to send or not send before inventory or other operation. Example (not to send select before inventory):

Header	Type	Command	PL(MSB)	PL(LSB)	Mode	Checksum	End
AA	00	12	00	01	01	14	DD

Type: 0x00
Command: 0x12
PL: 0x0001
Mode: 0x01
Checksum: 0x14

The meaning of select Mode：

0x00: Send select before all tag operation to select a certain tag.
0x01: Not to send select before all tag operation.
0x02: Send select before all tag operation except Inventory (Single Inventory and Multiple Inventory).E.g. select a certain tag before Read, Write, Lock, and Kill.

4.6.2. Response frame

When the select is set to be send or not send successfully, a response frame will be return:

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

Type: 0x01
Command: 0x0C
PL: 0x0001
Data: 0x00(success)
Checksum: 0x0E

4.7. Read

4.7.1. Command frame

Read the memory bank of tag from the given address and by the given length. The unit for segment address (SA) and data length (DL) is word. Before this command, a set select parameter command should be sent first to select a signal tag. The access command will not be send if the access password is all zero.

Header	Type	Command	PL(MSB)	PL(LSB)	AP(MSB)		
AA	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	DD

Type: 0x00
Command: 0x39
PL 0x0009
Access Password: 0x0000FFFF
MemBank: 0x03(User Memory Bank)
Segment Address: 0x0000
Data Length: 0x0002
Checksum: 0x45

4.7.2. Response frame

If the required data is read successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	UL	PC(MSB)	PC(LSB)
AA	01	39	00	13	0E	34	00
EPC(MSB)							
30	75	1F	EB	70	5C	59	04
			EPC(LSB)	**Data(MSB)**			**Data(LSB)**
E3	D5	0D	70	12	34	56	78
Checksum	**End**						
B0	DD						

Type: 0x01
Command: 0x39
PL: 0x0013
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Data: 0x12345678
Checksum: 0x52

If there is no tag response, an error code of 0x09 will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Error Code	Checksum	End
AA	01	FF	00	01	09	0A	DD

Type: 0x01
Command: 0xFF
PL: 0x0001
Error Code: 0x09
Checksum: 0x0A

If the access password is not correct, an error code of 0x16 will be returned, and the tag's PC+EPC will be sent back:

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

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0x16
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

If there are errors in SA or DL (e.g. the reading address is exceed the data memory bound), an error code of 0xA3 will be returned, and the tag's PC+EPC will be sent back:

Header	Type	Command	PL(MSB)	PL(LSB)	Error Code	UL	PC(MSB)
AA	01	FF	00	10	A3	0E	34
PC(LSB)	**EPC(MSB)**						
00	30	75	1F	EB	70	5C	59
				EPC(LSB)	**Checksum**	**End**	
04	E3	D5	0D	70	02	DD	

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0xA3
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x02

4.8. Write

4.8.1. Command frame

Write the memory bank of tag from the given address and by the given length. The unit for segment address (SA) and data length (DL) is word. Before this command, a set select parameter command should be sent first to select a signal tag. The access command will not be send if the access password is all zero. The data length to be written cannot exceed 32 words (64 bytes).

Header	Type	Command	PL(MSB)	PL(LSB)	AP(MSB)		
AA	00	49	00	0D	00	00	FF
AP(LSB)	**MemBank**	**SA(MSB)**	**SA(LSB)**	**DL(MSB)**	**DL(LSB)**	**DT(MSB)**	
FF	03	00	00	00	02	12	34
	DT(LSB)	**Checksum**	**End**				
56	78	6D	DD				

Type: 0x00
Command: 0x39
PL: 0x000D
Access Password: 0x0000FFFF
MemBank: 0x03
Segment Address: 0x0000
Data Length: 0x0004
Data: 0x12345678
Checksum: 0x6D

4.8.2. Responds frame

If the data is written successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	UL	PC(MSB)	PC(LSB)
AA	01	49	00	10	0E	34	00
EPC(MSB)							
30	75	1F	EB	70	5C	59	04
			EPC(LSB)	**Parameter**	**Checksum**	**End**	
E3	D5	0D	70	00	A9	DD	

Type: 0x01
Command: 0x49
PL: 0x0010
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Parameter: 0x00 (execute success)
Checksum: 0xA9

If there is no tag response, an error code of 0x10 will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Error Code	Checksum	End
AA	01	FF	00	01	10	0A	DD

Type: 0x01

Command: 0xFF
PL: 0x0001
Error Code: 0x10
Checksum: 0x0A

If the access password is not correct, an error code of 0x16 will be returned, and the tag's PC+EPC will be sent back:

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

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0x16
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

If there is an error code according to EPC Gen2 Protocol occurs (e.g. the address exceeds the data memory bound, an error code of 0xB3 will be returned), the error code bitwise OR 0xB0 will be returned, and the tag's PC+EPC will be sent back:

Header	Type	Command	PL(MSB)	PL(LSB)	Error Code	UL	PC(MSB)
AA	01	FF	00	10	B3	0E	34
PC(LSB)	**EPC(MSB)**						
00	30	75	1F	EB	70	5C	59
				EPC(LSB)	**Checksum**	**End**	
04	E3	D5	0D	70	12	DD	

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0xB3
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x12

The meaning of error code:
0xB0: Other Error
0xB3: Memory Overrun
0xB4: Memory Locked
0xAA: Insufficient Power

4.9. Lock

4.9.1. Command frame

Lock or unlock the tag memory band by access password. Before this command, a set select parameter command should be sent first to select a signal tag.

Header	Type	Command	PL(MSB)	PL(LSB)	AP(MSB)		
AA	00	82	00	07	00	00	FF
AP(LSB)	**LD(MSB)**		**LD(LSB)**	**Checksum**	**End**		
FF	02	00	80	09	DD		

Type: 0x00
Command: 0x82
PL: 0x0007
Access Password: 0x0000FFFF
Lock payload: 0x020080
Checksum: 0x09

The 4bits MSB of the lock payload (LD) is reserved. And the next 20 bits is for lock mask (10 bits) and lock action (10 bits) accordingly.

The detailed lock payload, masks and action field should refer to EPC^{TM} Radio-Frequency Identity Protocols from EPC global

4.9.2. Response frame

If lock operation is successes, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	UL	PC(MSB)	PC(LSB)
AA	01	82	00	10	0E	34	00
EPC(MSB)							
30	75	1F	EB	70	5C	59	04
			EPC(LSB)	**Parameter**	**Checksum**	**End**	
E3	D5	0D	70	00	E2	DD	

Type: 0x01
Command: 0x82
PL: 0x0010
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Parameter: 0x00
Checksum: 0xE2

If there is no tag response, an error code of 0x13 will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Error Code	Checksum	End
AA	01	FF	00	01	13	14	DD

Type: 0x01
Command: 0xFF
PL: 0x0001

Error Code: 0x13
Checksum: 0x14

If the access password is not correct, an error code of 0x16 will be returned, and the tag's PC+EPC will be sent back:

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

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0x16
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

If there is an error code according to EPC Gen2 Protocol occurs (e.g. the Memory Bank has been permalocked, an error code of 0xC4 will be returned), the error code bitwise OR 0xC0 will be returned, and the tag's PC+EPC will be sent back:

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

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0xC4
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x23

The meaning of error code:
0xC0: Other Error
0xC3: Memory Overrun
0xC4: Memory Locked
0xCB: Insufficient Power

4.10. Kill

4.10.1. Command frame

Before this command, a set select parameter command should be sent first to select a signal tag.

The kill command is as follow:

Header	Type	Command	PL(MSB)	PL(LSB)	KP(MSB)		
AA	00	65	00	04	00	00	FF
KP(LSB)	**Checksum**	**End**					
FF	67	DD					

Type: 0x00

Command: 0x65

PL: 0x0004

Kill Password: 0x0000FFFF

Checksum: 0x67

Notice: A tag with all zero Kill Password cannot be killed!

4.10.2. Response frame

If kill operation is successes and the tag responded CRC is correct, a response frame will be returned:

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

Type: 0x01

Command: 0x65

PL: 0x0010

PC+EPC length: 0x0E

PC: 0x3400

EPC: 0x30751FEB705C5904E3D50D70

Parameter: 0x00

Checksum: 0xC5

If there is no tag response, an error code of 0x12 will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Error Code	Checksum	End
AA	01	FF	00	01	12	13	DD

Type: 0x01

Command: 0xFF

PL: 0x0001

Error Code: 0x12

Checksum: 0x13

If there is an error code according to EPC Gen2 Protocol occurs (e.g. the tag has no kill password, an error code of 0xD0 will be returned), the error code bitwise OR 0xD0 will be returned, and the tag's PC+EPC will be sent back:

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

Type: 0x01
Command: 0xFF
PL: 0x0001
Error Code: 0xD0
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x2F

4.11. Get Query Parameters

4.11.1. Command frame

To get query parameters:

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End	
AA	00	0D	00	00	0D	DD	

Type: 0x00
Command: 0x0D
PL: 0x0000
Checksum: 0x0D

4.11.2. Response frame

If get query parameters successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Para(MSB)	Para(LSB)	Checksum
AA	01	0D	00	02	10	20	40
End							
DD							

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

In the above example, the query parameters are as follows:
DR=8, M=1, TRext=Use pilot tone, Sel=00, Session=00, Target=A, Q=4

where:
DR (1 bit): DR=8(1'b0), DR=64/3(1'b1). Only DR=8 is supported
M (2 bit): M=1(2'b00), M=2(2'b01), M=4(2'b10), M=8(2'b11). Only M=1 is supported.
TRext (1 bit): No pilot tone(1'b0), Use pilot tone(1'b1). Only use pilot tone (1'b1) 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

4.12. Set Query Parameters

4.12.1. Command frame

Set parameters of Query. The parameters (2 byte length) are defined as follows:

DR(1 bit): DR=8(1'b0), DR=64/3(1'b1). Only DR=8 is supported

M(2 bit): M=1(2'b00), M=2(2'b01), M=4(2'b10), M=8(2'b11). Only M=1 is supported

TRext(1 bit): No pilot tone(1'b0), Use pilot tone(1'b1). Only use pilot tone(1'b1) 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

Example (DR=8, M=1, TRext=Use pilot tone, Sel=00, Session=00, Target=A, Q=4):

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

Type: 0x00

Command: 0x0E

PL: 0x0002

Parameter: 0x1020

Checksum: 0xC6

4.12.2. Response frame

If set query parameters successfully, a response frame will be returned:

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

Type: 0x01

Command: 0x0E

PL: 0x0001

Parameter: 0x00

Checksum: 0x10

4.13. Set Region

4.13.1. Command frame

Set region for radio frequency regulation. Example (China band):

Header	Type	Command	PL(MSB)	PL(LSB)	Region	Checksum	End
AA	00	07	00	01	01	09	DD

Type: 0x00
Command: 0x07
PL: 0x0001
Region: 0x01
Checksum: 0x09

Region Code Table：

Region	Parameter
China 900MHz	01
China 800MHz	04
USA	02
Europe	03
Korea	06

4.13.2. Response frame

If region is set successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	07	00	01	00	09	DD

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

4.14. Get Region

4.14.1. Command frame

Get module current region command.

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	08	00	00	08	DD

Type: 0x00
Command: 0x08
PL: 0x0000
Checksum: 0x08

4.14.2. Response frame

A response frame includes current working region (in this example, region is China 900MHz) will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Region	Checksum	End
AA	01	08	00	01	01	0B	DD

Type: 0x01
Command: 0x08
PL: 0x0001
Region: 0x01
Checksum: 0x0B

4.15. Set Channel

4.15.1. Command frame

Set the current working channel. Example (In China band, set the working channel to 920.125MHz)

Header	Type	Command	PL(MSB)	PL(LSB)	CH Index	Checksum	End
AA	00	AB	00	01	01	AC	DD

Type: 0x00
Command: 0xAB
PL: 0x0001
Channel Index: 0x01
Checksum: 0xAC

Calculate channel index from channel central frequency (Freq_CH is the channel central frequency):
China 900MHz band channel index:
CH_Index = (Freq_CH-920.125M)/0.25M

China 800MHz band channel index:
CH_Index = (Freq_CH-840.125M)/0.25M

Amercian band channel index:
CH_Index = (Freq_CH-902.25M)/0.5M

European band channel index:
CH_Index = (Freq_CH-865.1M)/0.2M

Korean band channel index:
CH_Index = (Freq_CH-917.1M)/0.2M

4.15.2. Response frame

If set channel successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	AB	00	01	00	AD	DD

Type: 0x01
Command: 0xAB
长度 PL: 0x0001
Parameter: 0x00
Checksum: 0xAD

4.16. Get Channel

4.16.1. Command frame

Get the current working channel:

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	AA	00	00	AA	DD

Type: 0x00
Command: 0xAA
PL: 0x0000
Checksum: 0xAA

4.16.2. Response frame

If get channel successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	AA	00	01	00	AC	DD

Type: 0x01
Command: 0xAA
PL: 0x0001
Parameter: 0x00(Channel_Index: 0x00)
Checksum: 0xAC

Calculate channel frequency from channel index:
China 900MHz band:
Freq_CH = CH_Index * 0.25M + 920.125M

China 800MHz band:
Freq_CH = CH_Index * 0.25M + 840.125M

American band:
Freq_CH = CH_Index * 0.5M + 902.25M

European band:
Freq_CH = CH_Index * 0.2M + 865.1M

Korean band:
Freq_CH = CH_Index * 0.2M + 917.1M

4.17. Frequency Hopping

4.17.1. Command frame

Enable/Disable frequency hopping:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	00	AD	00	01	FF	AD	DD

Type: 0x00
Command: 0xAD
PL: 0x0001
Parameter: 0xFF(0xFF: enable frequency hopping; 0x00: disable frequency hopping)
Checksum: 0xAD

4.17.2. Response frame

If frequency hopping is set successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	AD	00	01	00	AF	DD

Type: 0x01
Command: 0xAD
PL: 0x0001
Parameter: 0x00
Checksum: 0xAF

4.18. Get TX Power

4.18.1. Command frame

Get the current transmitting power:

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	B7	00	00	B7	DD

Type: 0x00
Command: 0xB7
PL: 0x0000
Checksum: 0xB7

4.18.2. Response frame

If get TX power successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Pow(MSB)	Pow(LSB)	Checksum
AA	01	B7	00	02	07	D0	91
End							
DD							

Type: 0x01
Command: 0xB7
PL: 0x0002
Power: 0x07D0(in the unit of dBm. e.g. 0x07D0_Hex = 2000_Dec: 20dBm)
Checksum: 0x91

4.19. Set TX Power

4.19.1. Command frame

Set current transmitting power:

Header	Type	Command	PL(MSB)	PL(LSB)	Pow(MSB)	Pow(LSB)	Checksum
AA	00	B6	00	02	07	D0	8F
End							
DD							

Type: 0x00
Command: 0xB6
PL: 0x0002
Pow: 0x07D0(in the unit of dBm)
Checksum: 0x8F

4.19.2. Response frame

The response frame is as follow:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	B6	00	01	00	B8	DD

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

4.20. Continuous Wave

4.20.1. Command frame

On/Off continuous wave output:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	00	B0	00	01	FF	B0	DD

Type: 0x00
Command: 0xB0
PL: 0x0001
Parameter: 0xFF(0xFF: on; 0x00: off)
Checksum: 0xB0

4.20.2. Response frame

If set continuous wave successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	B0	00	01	00	B2	DD

Type: 0x01
Command: 0xB0
PL: 0x0001
Parameter: 0x00
Checksum: 0xB2

4.21. Get Modem Parameters

4.21.1. Command frame

To get the current key modem parameters that includes: the mixer gain, the intermediate frequency (baseband) amplifier gain and the threshold for demodulation (the tag returned signal with RSSI higher than threshold will be demodulated):

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	F1	00	00	F1	DD

Type: 0x00
Command: 0xF1
PL: 0x0000
Checksum: 0xF1

4.21.2. Response frame

If get modem parameters successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Mixer_G	IF_G	Thrd(MSB)
AA	01	F1	00	04	03	06	01
Thrd(LSB)	**Checksum**	**End**					
B0	B0	DD					

Type: 0x01
Command: 0xF1
PL: 0x0004
Mixer_G: 0x03(Mixer gain: 9dB)
IF_G: 0x06(IF AMP gain: 36dB)
Threshold: 0x01B0 (The smaller the threshold is, the weaker tag returned signal could be received, but with the higher package error rate. So small threshold is for short communication, like inventory, to get longer reading distance. And large threshold is recommended for long communication, like read/write, to increase the success rate. 0x01B0 is a typical small threshold)
Checksum: 0xB0

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

4.22. Set Modem Parameters

4.22.1. Command frame

T o set the key modem parameters which includes: the mixer gain, the IF amplifier gain and the threshold for demodulation:

Header	Type	Command	PL(MSB)	PL(LSB)	Mixer_G	IF_G	Thrd(MSB)
AA	00	F0	00	04	03	06	01
Thrd(LSB)	**Checksum**	**End**					
B0	AE	DD					

Type: 0x00

Command: 0xF0

PL: 0x0004

Mixer_G: 0x03(Mixer gain: 9dB)

IF_G: 0x06(IF AMP gain: 36dB)

Threshold: 0x01B0(The smaller the threshold is, the weaker tag returned signal could be received, but with the higher package error rate. So small threshold is for short communication, like inventory, to get longer reading distance. And large threshold is recommended for long communication, like read/write, to increase the success rate. 0x01B0 is a typical small threshold)

Checksum: 0xAE

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

4.22.2. Response frame

If set modem parameters successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	F0	00	01	00	F2	DD

Type: 0x01

Command:	0xF0
PL:	0x0001
Parameter:	0x00
Checksum:	0xF1

4.23. Scan Jammer

4.23.1. Command frame

To scan the jammer signal strength at the RF input port of the reader chip (a continuous wave will be transmitted at the current set power when doing the scan). The channel to be scan is determined by the current frequency band setting.

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	F2	00	00	F2	DD

Type: 0x00
Command: 0xF2
PL: 0x0000
Checksum: 0xF2

4.23.2. Response frame

The scan jammer result will be returned by a response frame. E.g. in China 900MHz band, there are 20 channels. If the scan jammer is executed successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	CH_L	CH_H	JMR(MSB)
AA	01	F2	00	16	00	13	F2
F1	F0	EF	EC	EA	E8	EA	EC
EE	F0	F1	F5	F5	F5	F6	F5
		JMR(LSB)	Checksum	End			
F5	F5	F5	DD	DD			

Type: 0x01
Command: 0xF2
PL: 0x0016
CH_L: 0x00 (the first channel index is: 0)
CH_H: 0x13 (the last channel index is: 19)
JMR: 0xF2F1F0EFECEAE8EAECEEF0F1F5F5F5F6F5F5F5F5 (0xF2: -14dBm)
Checksum: 0xDD

4.24. Scan Channel (LBT)

4.24.1. Command frame

Scan the channel power at the RF input power of the reader chip:

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	F3	00	00	F3	DD

Type: 0x00
Command: 0xF3
PL: 0x0000
Checksum: 0xF3

4.24.2. Response frame

The scan channel result will be returned by a response frame. E.g. in China 900MHz band, there are 20 channels. If the scan channel is executed successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	CH_L	CH_H	RSSI(MSB)
AA	01	F3	00	16	00	13	BA
BA	BA	BA	BA	BA	BA	BA	BA
BA	BA	BA	BA	BA	BA	BA	BA
		RSSI(LSB)	Checksum	End			
BA	BA	BA	A5	DD			

Type: 0x01
Command: 0xF2
PL: 0x0016
CH_L: 0x00(the first channel index is: 0)
CH_H: 0x13(the last channel index is: 19)
JMR: 0xBABABABABABABABABABABABABABABABABABABABABA(0xBA: -70dBm, is the lower limit of the RSSI)
Checksum: 0xDD

4.25. GPIO

4.25.1. Command frame

Set the IO direction, set/get IO status:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter0	Parameter1	Parameter2
AA	01	1A	00	03	00	04	01
Checksum	**End**						
22	DD						

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

Parameters:

Index	Description	Length	Operation
0	Parameter0	1 byte	Operation: 0x00：Set IO direction; 0x01：Set IO status； 0x02：Get IO status; Current IO is defined by parameter1;
1	Parameter1	1 byte	Range: 0x01-0x04: IO1-IO4
2	Parameter2	1 byte	Range: 0x00-0x01。 Parameter0 | Parameter2 | Description 0x00 | 0x00 | IO direction: Input 0x00 | 0x01 | IO direction: Output 0x01 | 0x00 | IO status: Low 0x01 | 0x01 | IO status: High When Parameter0=0x02, Parameter2 will be ignored.

4.25.2. Response frame

If the IO is set successfully, a response frame will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter0	Parameter1	Parameter2
AA	01	1A	00	03	00	04	01
Checksum	**End**						
23	DD						

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

<table>
<tr><th>Index</th><th>Description</th><th>Length</th><th>说明</th></tr>
<tr><td>0</td><td>Parameter0</td><td>1 byte</td><td>Operation:
0x00：Set IO direction;
0x01：Set IO status；
0x02：Get IO status;
Current IO is defined by parameter1;</td></tr>
<tr><td>1</td><td>Parameter 1</td><td>1 byte</td><td>Range: 0x01-0x04: IO1-IO4</td></tr>
<tr><td>2</td><td>Parameter 2</td><td>1 byte</td><td>Range: 0x00-0x01。
<table>
<tr><th>Parameter0</th><th>Parameter2</th><th>Description</th></tr>
<tr><td>0x00</td><td>0x00</td><td>Set IO direction fail</td></tr>
<tr><td>0x00</td><td>0x01</td><td>Set IO direction success</td></tr>
<tr><td>0x01</td><td>0x00</td><td>Set IO status fail</td></tr>
<tr><td>0x01</td><td>0x01</td><td>Set IO status success</td></tr>
<tr><td>0x02</td><td>0x00</td><td>IO status is low</td></tr>
<tr><td>0x02</td><td>0x01</td><td>IO status is high</td></tr>
</table></td></tr>
</table>

4.26. Module Sleep

4.26.1. Command frame

The Module Sleep command can make the module stay a low power status. Any byte sent by UART can wake up the module and the byte will be discarded. This command will also reset the chip. The firmware will be downloaded to chip after waking up. So some parameters might be reset.

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	17	00	00	17	DD

Type: 0x00
Command: 0x17
PL: 0x0000
Checksum: 0x17

4.26.2. Response frame

The response frame is defined as blow:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	17	00	01	00	19	DD

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

4.27. Set Module Idle Time

4.27.1. Command frame

This command can set how many minutes the module stays idle before it enters into sleep mode automatically.

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	00	1D	00	01	02	20	DD

Type: 0x00
Command: 0x1D
PL: 0x0001
Parameter: 0x02(sleep after 2 minutes no operation，range 1~9minutes，0x00 means no automatic sleep)
Checksum: 0x20

4.27.2. Response frame

The response frame is defined as blow：

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	1D	00	01	02	21	DD

Type: 0x01
Command: 0x1D
PL: 0x0001
Parameter: 0x02（Idle time）
Checksum: 0x21

4.28. NXP ReadProtect/Reset ReadProtect

NXP ReadProtect/Reset ReadProtect

NXP G2X tags support ReadProtect/Reset ReadProtect commands. The tag's ProtectEPC and ProtectTID bits will be set to '1' after executing ReadProtect successfully, and the tag will be in data protection status. A Reset ReadProtect command should be executed if a user wants the tag return to normal status. Before this command, a set select parameter command should be sent first to select a signal tag.

4.28.1. Command frame

ReadProtect/Reset ReadProtect command is defined as blow:

Header	Type	Command	PL(MSB)	PL(LSB)	AP(MSB)		
AA	00	E1	00	05	00	00	FF
AP(LSB)	**Reset**	**Checksum**	**End**				
FF	00	E4	DD				

Type: 0x00
Command: 0xE1
PL: 0x0005
Access Password: 0x0000FFFF
ReadProtect/Reset ReadProtect: 0x00(0x00: ReadProtect，0x01: Reset ReadProtect)
Checksum: 0x0B

4.28.2. Response frame

If a ReadProtect executing successfully，the response frame is：

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

Type: 0x01
Command: 0xE1
PL: 0x0010
PC+EPC length UL: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Parameter: 0x00(executing success)
Checksum: 0x3D

If a Reset ReadProtect executing successfully，the response frame is：

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

Type: 0x01

Command: 0xE2
PL: 0x0010
PC+EPC lengthUL: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Parameter: 0x00(executing success)
Checksum: 0x3E

If there is no tag response after ReadProtect (Set/Reset parameter is 0x00) command, an error code of 0x2A will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	FF	00	01	2A	2B	DD

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

If there is no tag response after Reset ReadProtect (Set/Reset parameter is 0x01) command, an error code of 0x2B will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	FF	00	01	2B	2C	DD

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

If the access password is not correct, an error code of 0x16 will be returned, and the tag's PC+EPC will be sent back:

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

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0x16
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

4.29. NXP Change EAS

NXP G2X tags support Change EAS command. The tag's PSF bit will be set to '1' or '0' after executing Change EAS command. The tag will answer EAS_Alarm command when PSF bit is '1', otherwise the tag will not answer EAS_Alarm command when PSF bit is '0'. Before this command, a set select parameter command should be sent first to select a signal tag.

4.29.1. Command frame

Change EAS command is defined as blow:

Header	Type	Command	PL(MSB)	PL(LSB)	AP(MSB)		
AA	00	E3	00	05	00	00	FF
AP(LSB)	**PSF**	**Checksum**	**End**				
FF	01	E7	DD				

Type: 0x00
Command: 0xE3
PL: 0x0005
Access Password: 0x0000FFFF
Set/Reset: 0x01(0x01 means set PSF bit to '1', 0x00 means set PSF to '0')
Checksum: 0xE7

4.29.2. Response frame

If Change EAS command executing successfully，the response frame is:

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

Type: 0x01
Command: 0xE3
PL: 0x0010
PC+EPC length UL: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Parameter: 0x00(success)
Checksum: 0x3F

If there is no tag response after Change EAS command, an error code of 0x1B will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	FF	00	01	1B	1C	DD

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

If the access password is not correct, an error code of 0x16 will be returned, and the tag's PC+EPC will be sent back:

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

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0x16
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

4.30. NXP EAS_Alarm

NXP G2X tags support EAS_Alarm command. The tag will answer 64bits EAS-Alarm code immediately after EAS-alarm command. The tag will answer EAS-Alarm command only when PSF bit is set to '1'.

4.30.1. Command frame

EAS_Alarm command is defined as blow:

Header	Type	Command	PL(MSB)	PL(LSB)	Checksum	End
AA	00	E4	00	00	E4	DD

Type: 0x00
Command: 0xE4
PL: 0x0000
Checksum: 0xE4

4.30.2. Response frame

If EAS_Alarm command executing successfully and the tag answers the right 64bits EAS-Alarm code, the response frame is:

Header	Type	Command	PL(MSB)	PL(LSB)	EAS-Alarm code(MSB)		
AA	01	E4	00	08	69	0A	EC
				EAS-Alarm code(LSB)	Checksum	End	
7C	D2	15	D8	F9	80	DD	

Type: 0x01
Command: 0xE3
PL: 0x0001
Parameter: 0x00
Checksum: 0xE5

If there is no tag response after Change EAS command, an error code of 0x1D will be returned:

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

Type: 0x01
Command: 0xFF
PL: 0x0001
Parameter: 0x1D
Checksum: 0x1E

4.31. NXP ChangeConfig 指令

Some series of NXP G2X tags such as G2iM and G2iM+ support ChangeConfig command. This command can read or modify the 16bits Config-Word. The Config-Word is stored at address 20h of Bank 01 (EPC MemBank). It can be read or write by standard READ/WRITE command. The bits to be toggled in the configuration register need to be set to '1'. Before this command, a set select parameter command should be sent first to select a signal tag.

4.31.1. Command frame

ChangeConfig command is defined as blow:

Header	Type	Command	PL(MSB)	PL(LSB)	AP(MSB)		
AA	00	E0	00	06	00	00	FF
AP(LSB)	**Config(MSB)**	**Config (LSB)**	**Checksum**	**End**			
FF	00	00	E4	DD			

Type: 0x00
Command: 0xE0
PL: 0x0006
Access Password: 0x0000FFFF
Config-Word: 0x0000(all 0 means response is unchanged Config-Word, same as read)
Checksum: 0xE4

4.31.2. Response frame

If ChangeConfig executing successfully, the response is:

Header	Type	Command	PL(MSB)	PL(LSB)	UL	PC(MSB)	PC(LSB)
AA	01	E0	00	11	0E	30	00
EPC(MSB)							
30	75	1F	EB	70	5C	59	04
			EPC(LSB)	**Config(MSB)**	**Config(LSB)**	**Checksum**	**End**
E3	D5	0D	70	00	41	DD	DD

Type: 0x01
Command: 0xE0
PL: 0x0011
UL: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Config-Word: 0x0041
Checksum: 0xDD

If there is no tag response after ChangeConfig command, an error code of 0x1A will be returned:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	FF	00	01	1A	1B	DD

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

If the access password is not correct, an error code of 0x16 will be returned, and the tag's PC+EPC will be sent back:

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

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0x16
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

4.32. Impinj Monza QT

Impinj Monza 4QT tags support QT command that can modify the QT Control Word. Setting the QT_SR bit can reduce the tag's operation range and modifying the QT_MEM bit can switch the tag to use Public Memory Map or Private Memory Map. Before this command, a set select parameter command should be sent first to select a signal tag.

4.32.1. Command frame

QT command is defined as blow (set QT_MEM to 1 and write it to Non-volatilizing memory in this example):

Header	Type	Command	PL(MSB)	PL(LSB)	AP(MSB)		
AA	00	E5	00	08	00	00	FF
AP(LSB)	**Read/Write**	**Persistence**	**Payload0**	**Payload1**	**Checksum**	**End**	
FF	01	01	40	00	2D	DD	

Type: 0x00
Command: 0xE5
PL: 0x0008
Access Password: 0x0000FFFF
Read/Write: 0x01(0x00: Read，0x01: Write)
Persistence: 0x01(0x00: write to volatilizing memory, 0x01: write it to Non-volatilizing memory, should always be 0x01)
Payload: 0x4000(QT Control，two MSB bits is QT_SR and QT_MEM)
Checksum: 0x2D

4.32.2. Response frame

If QT command executing successfully and Read/Write field is 0x00, the response frame is:

Header	Type	Command	PL(MSB)	PL(LSB)	UL	PC(MSB)	PC(LSB)
AA	01	E5	00	11	0E	30	00
EPC(MSB)							
30	75	1F	EB	70	5C	59	04
			EPC(LSB)	**QT Control0**	**QT Control1**	**Checksum**	**End**
E3	D5	0D	70	00	00	42	DD

Type: 0x01
Command: 0xE5
PL: 0x0011
PC+EPC length UL: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
QT Control Word: 0x0000
Checksum: 0x42

If QT command executing successfully and Read/Write field is 0x01, the response frame is:

Header	Type	Command	PL(MSB)	PL(LSB)	UL	PC(MSB)	PC(LSB)
AA	01	E6	00	10	0E	30	00
EPC(MSB)							
30	75	1F	EB	70	5C	59	04
			EPC(LSB)	**Parameter**	**Checksum**	**End**	

E3	D5	0D	70	00	42	DD	

Type: 0x01
Command: 0xE6
PL: 0x0010
PC+EPC length UL: 0x0E
PC: 0x3000
EPC: 0x30751FEB705C5904E3D50D70
Parameter: 0x00(success)
Checksum: 0x42

If there is no tag response after QT command, an error code of 0x2E will be returned:

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

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

If the access password is not correct, an error code of 0x16 will be returned, and the tag's PC+EPC will be sent back:

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

Type: 0x01
Command: 0xFF
PL: 0x0010
Error Code: 0x16
PC+EPC length: 0x0E
PC: 0x3400
EPC: 0x30751FEB705C5904E3D50D70
Checksum: 0x75

5. Summary

Code	Description
0x03	Get Module Information
0x22	Single Inventory
0x27	Multiple Inventory
0x28	Stop Multiple Inventory
0x0C	Set Select Parameters
0x12	Set Select Mode
0x39	Read
0x49	Write
0x82	Lock
0x65	Kill
0x0D	Get Query Parameters
0x0E	Set Query Parameters
0x07	Set Region
0x08	Get Region
0xAB	Set Channel
0xAA	Get Channel
0xAD	Frequency Hopping
0xB7	Get TX Power
0xB6	Set TX Power
0xB0	Continuous Wave
0xF1	Get Modem Parameters
0xF0	Set Modem Parameters
0xF2	Scan Jammer
0xF3	Scan Channel
0x1A	GPIO
0x17	Module Sleep
0x1D	Set Module Idle Time
0xE1	NXP ReadProtec/Reset ReadProtect
0xE3	NXP Change EAS
0xE4	NXP EAS-Alarm
0xE0	NXP ChangeConfig
0xE5	Impinj Monza 4 QT

6. Error Code

If a command execution is fail, a fail response frame will be returned which includes an error code to indicate the error type. E.g. if inventory command is fail by no tag response or CRC error, an error code "0x15" will be returned in a response frame:

Header	Type	Command	PL(MSB)	PL(LSB)	Parameter	Checksum	End
AA	01	FF	00	01	15	16	DD

Type: 0x01
Command: 0xFF (0xFF indicates command fail)
PL: 0x01
Parameter: 0x15 (error code)
Checksum: 0x16

Error code summary:

Type	Code	Description
Command Error	0x17	Command code error
FHSS Fail	0x20	Frequency hopping time out. All channel are occupied.
Inventory Fail	0x15	Inventory fail. No tag response or CRC error.
Access Fail	0x16	Access fail. May caused by password error.
Read Fail	0x09	Read fail. No tag response or CRC error.
Read Error	0xA0 | Error code	Read error. Error code (0xA0 | Error code) indication refer to Tag-error code in the next table.
Write Fail	0x10	Write fail. No tag response or CRC error.
Write Error	0xB0 | Error code	Write error. Error code (0xB0 | Error code) indication refer to Tag-error code in the next table.
Lock Fail	0x13	Lock fail. No tag response or CRC error.
Lock Error	0xC0 | Error code	Lock error. Error code (0xC0 | Error code) indication refer to Tag-error code in the next table.
Kill Fail	0x13	Kill fail. No tag response or CRC error.
Kill Error	0xD0 | Error code	Kill error. Error code (0xC0 | Error code) indication refer to Tag-error code in the next table.

EPC Gen2 protocol specified error code:

Tag error-code

Error-code Support	Error Code	Error code Name	Error Description
Error-specific	00000000_2	Other error	Other errors not described in this table.
	00000011_2	Memory overrun	The given memory area does not exist. Or the tag does not support the given EPC length, e.g. XPC.
	00000100_2	Memory locked	The given memory area is locked and/or permanent locked. And the lock status is not readable or not writeable.
	00001011_2	Insufficient power	Tag has no enough power to write.
Non-specific	00001111_2	Non-specific error	Tag does not support error-code.

NXP G2X tags' specific error code：

ChangeConfig Fail	0x1A	ChangeConfig failed. No tag response or CRC error.
ReadProtect Fail	0x2A	ReadProtect failed. No tag response or CRC error.
Reset ReadProtect Fail	0x2B	Reset ReadProtect failed. No tag response or CRC error.
Change EAS Fail	0x1B	Change EAS failed. No tag response or CRC error.
EAS_Alarm Fail	0x1D	EAS_Alarm failed. No correct Alarm Code response.
Specific commands error code	0xE0 | Error code	Specific commands error code. Error code (0xE0 | Error code) indication refer to Tag-error code in the EPC Gen2 protocol.

Impinj Monza QT tags' specific error code:

QT Fail	0x2E	QT failed. No tag response or CRC error.
Specific commands error code	0xE0 | Error code	Specific commands error code. Error code (0xE0 | Error code) indication refer to Tag-error code in the EPC Gen2 protocol.