



OEM-UHF Reader Module NEO2

Communication Protocol

iDTRONIC GmbH
Ludwig-Reichling-Straße 4
67059 Ludwigshafen
Germany/Deutschland

Issue 0.1
– 03. June 2025 –

Phone: +49 621 6690094-0
Fax: +49 621 6690094-9
E-Mail: info@idtronic.de
Web: idtronic.de

Subject to alteration without prior notice.
© Copyright iDTRONIC GmbH 2025
Printed in Germany

Contents

1	Internal Microcontroller Unit (MCU)	4
2	Command Frame	5
2.1	Command Frame Structure	5
2.2	Command Type	5
3	Command Definition	6
3.1	Commands Overview	6
3.2	Get Module Information (0x03).....	6
3.3	Single Inventory (0x22).....	7
3.4	Multiple Inventory (0x27).....	8
3.5	Stop Multiple Inventory (0x28).....	9
3.6	Set Select Parameters (0x0C).....	9
3.7	Set Select Mode (0x12).....	10
3.8	Read (0x39)	10
3.9	Write (0x49)	11
3.10	Lock (0x82)	13
3.11	Kill (0x65)	14
3.12	Get Query Parameters (0x0D).....	15
3.13	Set Query Parameters (0x0E).....	16
3.14	Set Region (0x07).....	16
3.15	Get Region (0x08)	17
3.16	Set Channel (0xAB)	17
3.17	Get Channel (0xAA)	18
3.18	Frequency Hopping (0xAD)	18
3.19	Get TX Power (0xB7).....	19
3.20	Set TX Power (0xB6)	19
3.21	Continuous Wave (0xB0)	20
3.22	Get Modem Parameters (0xF1).....	20
3.23	Set Modem Parameters (0xF0)	21
3.24	Scan Jammer (0xF2).....	22
3.25	Scan Channel LBT (0xF3)	22
3.26	GPIO (0x1A)	23
3.27	Module Sleep (0x17).....	24
3.28	Set Module Idle Time (0x1D)	24
3.29	NXP ReadProtect/Reset ReadProtect (0xE1).....	24
3.30	NXP Change EAS (0xE3)	26
3.31	NXP EAS_Alarm (0xE4)	27
3.32	NXP ChangeConfig (0xE0)	27
3.33	Impinj Monza 4QT (0xE5)	28
4	Error Code	30

1 Internal Microcontroller Unit (MCU)

The RFID Module has an internal 8-bit 8051 MCU, which has 256-byte RAM, 16 Kbyte program ROM and 3 timers in the firmware (timer 2 is used for baud rate control, timer 0 is used for frequency hopping control and timer 1 is reserved for the user). Additionally, there is an 8 Kbyte data RAM. In the receiving phase, the data RAM can not be accessed by the MCU. There is no internal flash and the firmware of the MCU can be downloaded using the UART or I2C interface.

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

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

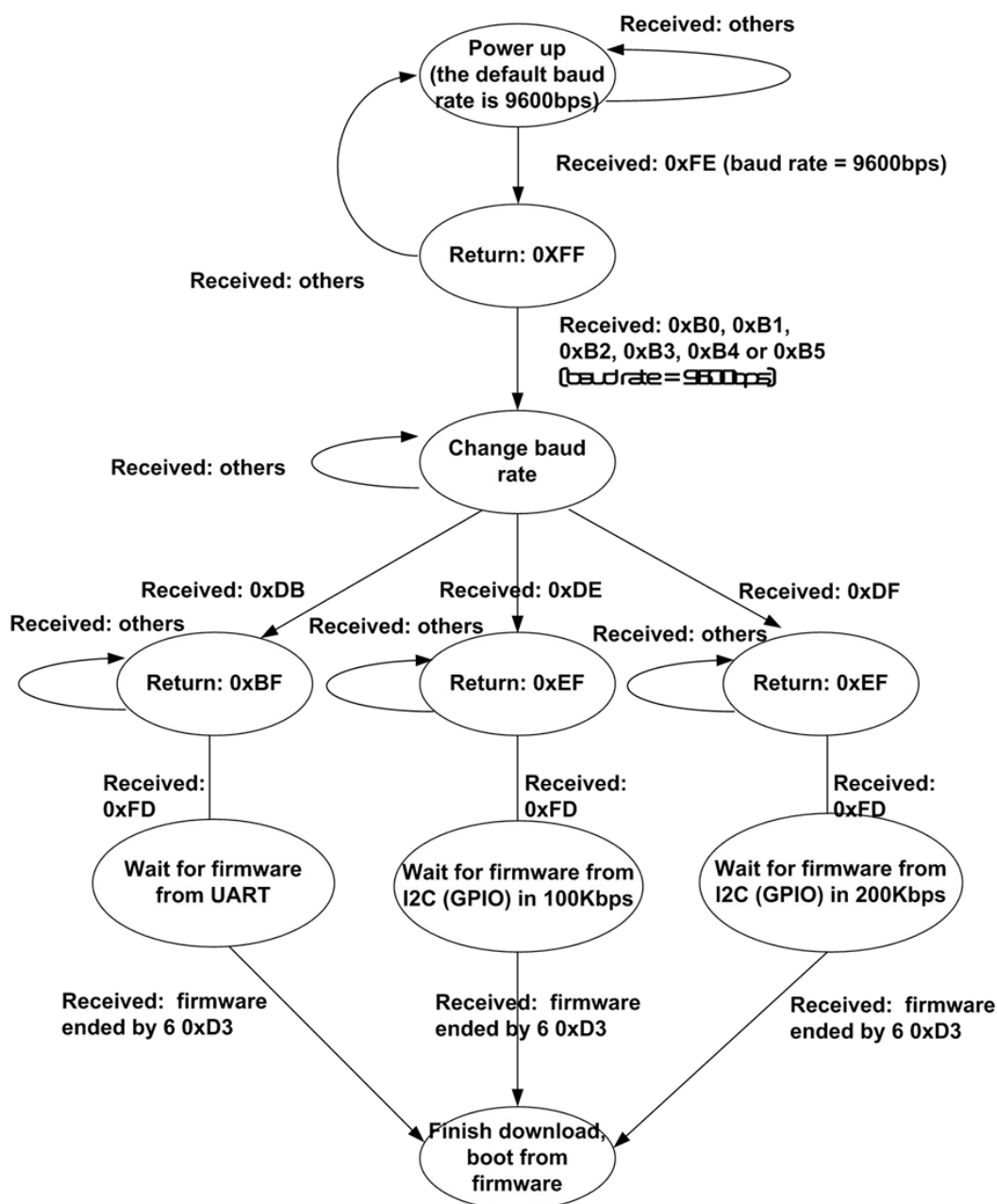


Figure 1: Handshake protocol for firmware download

Baud Rate Settings:

Parameter	Baud Rate (bps)
0xB0	9600
0xB1	19200
0xB2	28800

Parameter	Baud Rate (bps)
0xB3	38400
0xB4	57600
0xB5	115200

2 Command Frame

2.1 Command Frame Structure

Every command includes a Header, Command Type, Command Code, Parameter Length (PL), Checksum, End.

Example

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

The checksum is calculated by taking the least significant byte of the sum from Type to Parameter.

2.2 Command Type

Type	Description
0x00	Command frame: command received by firmware
0x01	Response frame: firmware response
0x02	Notification frame: firmware-initiated report

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

Notification frames are used for single or multiple inventories. The firmware sends a notification frame every time when a tag is read successfully.

3 Command Definition

3.1 Commands Overview

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 ReadProtect/Reset ReadProtect
0xE3	NXP Change EAS
0xE4	NXP EAS-Alarm
0xE0	NXP ChangeConfig
0xE5	Impinj Monza 4 QT

3.2 Get Module Information (0x03)

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

Command frame

Frame type	0x00	
Command Code	0x03	
Parameter	Hardware Version	0x00
	Software Version	0x01
	Manufacturer	0x02

Example: get hardware version.

>> BB 00 03 00 01 00 04 7E

Frame type	0x00
Command Code	0x03
PL	0x0001
Parameter	0x00
Checksum	0x04

Response frame

Frame type	0x01
Command Code	0x03
Data	Variable (ASCII codes)

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

The response to the example above is as follows:

<< BB 01 03 00 10 00 4D 31 30 30 20 32 36 64 42 6D 20 56 31 2E 30 22 7E

Frame type	0x01
Command Code	0x03
PL	0x0010
Info Type	0x00
Info	4D 31 30 30 20 32 36 64 42 6D 20 56 31 2E 30 ("M100 26dBm V1.0")
Checksum	0x22

3.3 Single Inventory (0x22)

This command triggers a single inventory operation. The power amplifier will be opened at the beginning of the inventory and closed after the inventory.

Command frame

>> BB 00 22 00 00 22 7E

Frame type	0x00
Command Code	0x22
PL	0x0000
Checksum	22

Notification frame

After the firmware received this command and successfully read a tag, a notification frame including RSSI, PC, EPC and CRC will be sent. If multiple tags are read in one inventory, the same number of notification frames as successfully read tags will be sent.

RSSI is a received signal strength indicator at the RF input port of the reader chip. It is a complementally coded signed hex in the unit of dBm. E.g. an RSSI of 0xC9 = -55 dBm.

<< BB 02 22 00 11 C9 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 3A 76 EF 7E

Frame type	0x02
Command Code	0x22
PL	0x0011
RSSI	0xC9
PC	0x3400
EPC	0x30751FEB705C5904E3D50D70
CRC	0x3A76

Checksum	0xEF
----------	------

Response frame

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

<< BB 01 FF 00 01 15 16 7E

Frame type	0x01
Command Code	0xFF
PL	0x0001
Parameter	0x15
Checksum	0x16

3.4 Multiple Inventory (0x27)

This command triggers multiple inventory operations. The number of inventories can be between 0 to 65535.

Command frame

Example: 10000 times inventory

>> BB 00 27 00 03 22 27 10 83 7E

Frame type	0x00
Command Code	0x27
PL	0x0003
Reserved	0x22
CNT	0x2710 (=10000 in decimal)
Checksum	0x83

Notification frame

The notification frame is the same as for single inventory.

<< BB 02 22 00 11 C9 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 3A 76 EF 7E

Frame type	0x02
Command Code	0x22
PL	0x0011
RSSI	0xC9
PC	0x3400
EPC	0x30751FEB705C5904E3D50D70
CRC	0x3A76
Checksum	0xEF

Response frame

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

<< BB 01 FF 00 01 15 16 7E

Frame type	0x01
Command Code	0xFF
PL	0x0001
Parameter	0x15
Checksum	0x16

3.5 Stop Multiple Inventory (0x28)

The multiple inventory can be stopped (not paused) immediately with this command.

Command frame

>> BB 00 28 00 00 28 7E

Frame type	0x00
Command Code	0x28
PL	0x0000
Checksum	0x28

Response frame

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

<< BB 01 28 00 01 00 2A 7E

Frame type	0x01
Command Code	0x28
PL	0x0001
Parameter	0x00
Checksum	0x2A

3.6 Set Select Parameters (0x0C)

This command will set “select parameters” and “Select Mode” to 0x02. The select will be send before inventory by default.

Command frame

Example: Select tags with the same values from bit 0x20 to bit 0x80.

>> BB 00 0C 00 13 01 00 00 00 20 60 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 AD 7E

Frame type	0x00
Command Code	0x0C
PL	0x0013
SelParam	0x01 (Target: 3'b000; Action: 3'b000; MemBank: 2'b01)
Ptr	0x00000020 (in bit) form 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-bit Target, 3-bit Action and 2-bit MemBank.

The values for MemBank are:

2'b00 – RFU Memory Bank

2'b01 – EPC Memory Bank

2'b10 – TID Memory Bank

2'b11 – User Memory Bank

For the meanings of Target and Action please refer to the EPC UHF Class 1 Gen 2 Standard Protocol.

Response frame

<< BB 01 0C 00 01 00 0E 7E

Frame type	0x01
Command Code	0x0C
PL	0x0001

Parameter	0x00
Checksum	0x0E

3.7 Set Select Mode (0x12)

After the command Set Select Parameters, the select command can be send or not send before inventory or other operations.

Command frame

Example: Don't send select before inventory.

>> BB 00 12 00 01 01 14 7E

Frame type	0x00
Command Code	0x12
PL	0x0001
Mode	0x01
Checksum	0x14

The values for select Mode are:

0x00 – Send select before all tag operations to select a certain tag

0x01 – Do not send select before all tag operations

0x02 – Send select before all tag operations except Inventory (Single and Multiple Inventory), e.g. select a tag before read, write, lock and kill

Response frame

When the select is set successfully, a response frame will be returned.

<< BB 01 12 00 01 00 14 7E

Frame type	0x01
Command Code	0x12
PL	0x0001
Data	0x00 (Success)
Checksum	0x14

3.8 Read (0x39)

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

Command frame

Example: Read the first two words (4 bytes) from the user memory bank

>> BB 00 39 00 09 00 00 FF FF 03 00 00 00 02 45 7E

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

Response frame

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

<< BB 01 39 00 13 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 12 34 56 78 B0 7E

Frame type	0x01
Command Code	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

<< BB 01 FF 00 01 09 0A 7E

Frame type	0x01
Command Code	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 and EPC will be sent

<< BB 01 FF 00 10 16 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 75 7E

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

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

<< BB 01 FF 00 10 A3 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 02 7E

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

3.9 Write (0x49)

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

Command frame

Example: Write 0x12345678 to the first two words (4 bytes) of the user memory bank

>> BB 00 49 00 0D 00 00 FF FF 03 00 00 00 02 12 34 56 78 6D 7E

Frame type	0x00
Command Code	0x49
PL	0x000D
Access Password	0x0000FFFF
MemBank	0x03
Segment Address	0x0000
Data Length	0x0002
Data	0x12345678
Checksum	0x6D

Response frame

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

<< BB 01 49 00 10 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 00 A9 7E

Frame type	0x01
Command Code	0x49
PL	0x0010
PC + EPC length	0x0E
PC	0x3400
EPC	0x30751FEB705C5904E3D50D70
Parameter	0x00 (executed successful)
Checksum	0xA9

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

<< BB 01 FF 00 01 10 0A 7E

Frame type	0x01
Command Code	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 and EPC will be sent

<< BB 01 FF 00 10 16 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 75 7E

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

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

<< BB 01 FF 00 10 B3 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 12 7E

Frame type	0x01
Command Code	0xFF

PL	0x0010
Error Code	0xB3
PC + EPC length	0x0E
PC	0x3400
EPC	0x30751FEB705C5904E3D50D70
Checksum	0x12

The meaning of the error code:

0xB0 – Other error

0xB3 – Memory Overrun

0xB4 – Memory Locked

0xBB – Insufficient Power

3.10 Lock (0x82)

Lock or unlock the memory bank of a tag using an access password. Before using this command, a set select parameter command should be sent first to select a single tag.

Command frame

Example:

>> BB 00 82 00 07 00 00 FF FF 02 00 80 09 7E

Frame type	0x00
Command Code	0x82
PL	0x0007
Access Password	0x0000FFFF
Lock payload	0x020080
Checksum	0x09

The 4 bits MSB of the lock payload are reserved and the next 20 bits are for the lock mask (10 bits) and the lock action (10 bits). For details on the lock payload please refer to the EPC Radio-Frequency Identity Protocols from EPC global.

Response frame

If the lock operation is successful, a response frame will be returned

<< BB 01 82 00 10 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 00 E2 7E

Frame type	0x01
Command Code	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

<< BB 01 FF 00 01 13 14 7E

Frame type	0x01
Command Code	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 and EPC will be sent

<< BB 01 FF 00 10 16 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 75 7E

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

If an error code according to the EPC Gen2 Protocol occurs (e.g. the memory bank has been permanently locked (0xC4)), the error code bitwise OR 0xC0 will be returned and the tag's PC and EPC will be sent

<< BB 01 FF 00 10 C4 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 23 7E

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

The meaning of the error code:

0xC0 – Other error

0xC3 – Memory Overrun

0xC4 – Memory Locked

0xCB – Insufficient Power

3.11 Kill (0x65)

Before using this command, the set select parameter command should be sent first to select a single tag.

Command frame

Example: Using kill password 0x0000FFFF

>> BB 00 65 00 04 00 00 FF FF 67 7E

Frame type	0x00
Command Code	0x65
PL	0x0004
Kill Password	0x0000FFFF
Checksum	0x67

Notice, a tag with a kill password all zero cannot be killed.

Response frame

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

<< BB 01 65 00 10 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 00 C5 7E

Frame type	0x01
Command Code	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

<< BB 01 FF 00 01 12 13 7E

Frame type	0x01
Command Code	0xFF
PL	0x0001
Error Code	0x12
Checksum	0x13

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

<< BB 01 FF 00 10 D0 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 2F 7E

Frame type	0x01
Command Code	0xFF
PL	0x0010
Error Code	0xD0
PC + EPC length	0x0E
PC	0x3400
EPC	0x30751FEB705C5904E3D50D70
Checksum	0x2F

3.12 Get Query Parameters (0x0D)

Command frame

>> BB 00 0D 00 00 0D 7E

Frame type	0x00
Command Code	0x0D
PL	0x0000
Checksum	0x0D

Response frame

If the operation was successful, a response frame will be returned

<< BB 01 0D 00 02 10 20 40 7E

Frame type	0x01
Command Code	0x0D
PL	0x0002
Query Parameter	0x1020
Checksum	0x40

In the above example, the query parameters are as follows:

DR = 8, M = 1, TReTx = Use pilot tone, Sel = 00, Session = 00, Target = A, Q = 4

Parameter	Length	Description
DR	1 bit	DR = 8 (1'b0); DR = 64/3 (1'b1). Only DR = 8 is supported
M	2 bits	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 bits	ALL (2'b00, 2'b01); Not 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 to 4'b1111

3.13 Set Query Parameters (0x0E)

Command frame

Example: DR = 8, M = 1, TRExt = Use pilot tone, Sel = 00, Session = 0, Target = A, Q = 4

>> BB 00 0E 00 02 10 20 40

Frame type	0x00
Command Code	0x0E
PL	0x0002
Query Parameter	0x1020
Checksum	0xC6

The parameters (2 bytes) are defined as follows:

Parameter	Length	Description
DR	1 bit	DR = 8 (1'b0); DR = 64/3 (1'b1). Only DR = 8 is supported
M	2 bits	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 bits	ALL (2'b00, 2'b01); Not 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 to 4'b1111

Response frame

If the operation was successful, a response frame will be sent

<< BB 01 0E 00 01 00 10 7E

Frame type	0x01
Command Code	0x0E
PL	0x0001
Parameter	0x00
Checksum	0x10

3.14 Set Region (0x07)

Set the region for radio frequency regulation.

Command frame

Example: Europe band

>> BB 00 07 00 01 03 0B 7E

Frame type	0x00
Command Code	0x07

PL	0x0001
Region	0x03
Checksum	0x0B

Region Code Table:

Region	Parameter
China 900 MHz	0x01
China 800 MHz	0x04
USA	0x02
Europe	0x03
Korea	0x06

Response frame

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

<< BB 01 07 00 01 00 09 7E

Frame type	0x01
Command Code	0x07
PL	0x0001
Parameter	0x00
Checksum	0x09

3.15 Get Region (0x08)

Get the current region of the module.

Command frame

>> BB 00 08 00 00 08 7E

Frame type	0x00
Command Code	0x08
PL	0x0000
Checksum	0x08

Response frame

<< BB 01 08 00 01 03 0D 7E

Frame type	0x01
Command Code	0x08
PL	0x0001
Region	0x03 (Europe)
Checksum	0x0D

3.16 Set Channel (0xAB)

Set the current working channel.

Command frame

Example: Set the working channel to 865.1 MHz in the european band

>>BB 00 AB 00 01 00 AC 7E

Frame type	0x00
Command Code	0xAB
PL	0x0001

Channel Index	0x00
Checksum	0xAC

Calculate the channel index from the channel central frequency (Freq_CH):

China 900 MHz band	$CH_Index = (Freq_CH - 920.125 \text{ MHz}) / 0.25 \text{ MHz}$
China 800 MHz band	$CH_Index = (Freq_CH - 840.125 \text{ MHz}) / 0.25 \text{ MHz}$
American band	$CH_Index = (Freq_CH - 902.25 \text{ MHz}) / 0.5 \text{ MHz}$
European band	$CH_Index = (Freq_CH - 865.1 \text{ MHz}) / 0.2 \text{ MHz}$
Korean	$CH_Index = (Freq_CH - 917.1 \text{ MHz}) / 0.2 \text{ MHz}$

Response frame

If the channel was set successfully, a response frame will be returned

<< BB 01 AB 00 01 00 AD 7E

Frame type	0x01
Command Code	0xAB
PL	0x0001
Parameter	0x00
Checksum	0xAD

3.17 Get Channel (0xAA)

Get the current channel.

Command frame

>> BB 00 AA 00 00 AA 7E

Frame type	0x00
Command Code	0xAA
PL	0x0000
Checksum	0xAA

Response frame

If the operation is successful, a response frame will be returned

<< BB 01 AA 00 01 00 AC 7E

Frame type	0x01
Command Code	0xAA
PL	0x0001
Channel Index	0x00
Checksum	0xAC

Calculate the channel frequency from the channel index:

China 900 MHz band	$Freq_CH = CH_Index * 0.25 \text{ MHz} + 920.125 \text{ MHz}$
China 800 MHz band	$Freq_CH = CH_Index * 0.25 \text{ MHz} + 840.125 \text{ MHz}$
American band	$Freq_CH = CH_Index * 0.5 \text{ MHz} + 902.25 \text{ MHz}$
European band	$Freq_CH = CH_Index * 0.2 \text{ MHz} + 865.1 \text{ MHz}$
Korean	$Freq_CH = CH_Index * 0.2 \text{ MHz} + 917.1 \text{ MHz}$

3.18 Frequency Hopping (0xAD)

Enable or disable frequency hopping.

Command frame

Example: enable frequency hopping

>> BB 00 AD 00 01 FF AD 7E

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

Response frame

If the operation was successful, a response frame will be returned

<< BB 01 AD 00 01 00 AF 7E

Frame type	0x01
Command Code	0xAD
PL	0x0001
Parameter	0x00
Checksum	0xAF

3.19 Get TX Power (0xB7)

Get the current transmitting power

Command frame

>> BB 00 B7 00 00 B7 7E

Frame type	0x00
Command Code	0xB7
PL	0x0000
Checksum	0xB7

Response frame

If the operation is successful, a response frame will be returned

<< BB 01 B7 00 02 07 D0 91 7E

Frame type	0x01
Command Code	0xB7
PL	0x0002
Power	0x07D0 (in dBm, e.g. 0x07D0 = 20 dBm)
Checksum	0x91

3.20 Set TX Power (0xB6)

Set the current transmitting power.

Command frame

Example: set to 20 dBm

>> BB 00 B6 00 02 07 D0 8F 7E

Frame type	0x00
Command Code	0xB6
PL	0x0002
Power	0x07D0 (in dBm, e.g. 0x07D0 = 20 dBm)
Checksum	0x8F

Response frame

<< BB 01 B6 00 01 00 B8 7E

Frame type	0x01
Command Code	0xB6
PL	0x0001
Checksum	0xB8

3.21 Continuous Wave (0xB0)

Enable or disable continuous wave output.

Command frame

Example: Continuous wave disabled

<< BB 00 B0 00 01 FF B0 7E

Frame type	0x00
Command Code	0xB0
PL	0x0001
Parameter	0xFF (0xFF: on; 0x00: off)
Checksum	0xB0

Response frame

<< BB 01 B0 00 01 00 B2 7E

Frame type	0x01
Command Code	0xB0
PL	0x0001
Parameter	0x00
Checksum	0xB2

3.22 Get Modem Parameters (0xF1)

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

Command frame

>> BB 00 F1 00 00 F1 7E

Frame type	0x00
Command Code	0xF1
PL	0x0000
Checksum	0xF1

Response frame

<< BB 01 F1 00 04 03 06 01 B0 B0 7E

Frame type	0x01
Command Code	0xF1
PL	0x0004
Mixer_G	0x03 (Mixer gain: 9 dB)
IF_G	0x06 (IF AMP gain: 36 dB)
Threshold	0x01B0
Checksum	0xB0

The smaller the threshold is, the weaker the received tag signal can be and the higher the package error rate. Hence, a small threshold is for short communication, like inventory, to get higher reading distance. A large threshold is recommended for long communication, like read/write, to increase the success rate. 0x01B0 is a typical small threshold.

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

3.23 Set Modem Parameters (0xF0)

Set the key modem parameters mixer gain, IF amplifier gain and the threshold for demodulation.

Command frame

Example: set mixer gain to 9 dB, the IF amplifier gain to 36 dB and the threshold to 0x01B0

>> BB 00 F0 00 04 03 06 01 B0 AE 7E

Frame type	0x00
Command Code	0xF0
PL	0x0004
Mixer_G	0x03
IF_G	0x06
Threshold	0x01B0
Checksum	0xAE

The smaller the threshold is, the weaker the received tag signal can be and the higher the package error rate. Hence, a small threshold is for short communication, like inventory, to get higher reading distance. A large threshold is recommended for long communication, like read/write, to increase the success rate. 0x01B0 is a typical small threshold.

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

Response frame

<< BB 01 F0 00 01 00 F2 7E

Frame type	0x01
Command Code	0xF0

3.26 GPIO (0x1A)

Set the IO direction and set or get IO status.

Command frame

>> BB 00 1A 00 03 00 04 01 22 7E

Frame type	0x00
Command Code	0x1A
PL	0x0003
Parameter	0x00 0x04 0x01
Checksum	0x22

Parameters:

Description	Length	Operation
Parameter0	1 byte	Operation: 0x00: Set IO direction 0x01: Set IO status 0x02: Get IO status
Parameter1	1 byte	Current IO, range 0x01 – 0x04
Parameter2	1 byte	Range 0x00 – 0x01 Depending on the value of Parameter 0 (0x00 or 0x01): 0x00 = Input or Low 0x01 = Output or High When Parameter0 is 0x02, Parameter2 will be ignored.

Response frame

<< BB 01 1A 00 03 00 04 01 23 7E

Frame type	0x01
Command Code	0x1A
PL	0x0003
Parameter	0x00 0x04 0x01
Checksum	0x23

Parameters:

Description	Length	Operation																		
Parameter0	1 byte	Operation: 0x00: Set IO direction 0x01: Set IO status 0x02: Get IO status																		
Parameter1	1 byte	Current IO, range 0x01 – 0x04																		
Parameter2	1 byte	Range 0x00 – 0x01 <table border="1"> <thead> <tr> <th>Parameter0</th><th>Parameter2</th><th>Description</th></tr> </thead> <tbody> <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> </tbody> </table>	Parameter0	Parameter2	Description	0x00	0x00	Set IO direction fail	0x00	0x01	Set IO direction success	0x01	0x00	Set IO status fail	0x01	0x01	Set IO status success	0x02	0x00	IO status is low
Parameter0	Parameter2	Description																		
0x00	0x00	Set IO direction fail																		
0x00	0x01	Set IO direction success																		
0x01	0x00	Set IO status fail																		
0x01	0x01	Set IO status success																		
0x02	0x00	IO status is low																		

		0x02	0x01	IO status is high
--	--	------	------	-------------------

3.27 Module Sleep (0x17)

This command signals the module to stay in a low power state. 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 the chip after waking up. Some parameters might be reset.

Command frame

>> BB 00 17 00 00 17 7E

Frame type	0x00
Command Code	0x17
PL	0x0000
Checksum	0x17

Response frame

<< BB 01 17 00 01 00 19 7E

Frame type	0x01
Command Code	0x17
PL	0x0001
Parameter	0x00
Checksum	0x19

3.28 Set Module Idle Time (0x1D)

Set how many minutes the module stays idle before it enters sleep mode automatically.

Command frame

>> BB 00 1D 00 01 02 20 7E

Frame type	0x00
Command Code	0x1D
PL	0x0001
Parameter	0x02 (sleep after 2 minutes no operation, range 1-9 minutes; 0x00 means no automatic sleep)
Checksum	0x20

Response frame

<< BB 01 1D 00 01 02 21 7E

Frame type	0x01
Command Code	0x1D
PL	0x0001
Parameter	0x02 (Idle time)
Checksum	0x21

3.29 NXP ReadProtect/Reset ReadProtect (0xE1)

NXP G2X tags support 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 to return to normal status. Before this command, the set select parameter command should be sent first to select a signal tag.

Command frame

>> BB 00 E1 00 05 00 00 FF FF 00 E4 7E

Frame type	0x00
Command Code	0xE1
PL	0x0005
Access Password	0x0000FFFF
ReadProtect	0x00 (0x00 = ReadProtect; 0x01 = Reset ReadProtect)
Checksum	0x0B

Response frame

ReadProtect:

<< BB 01 E1 00 10 0E 30 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 00 3D 7E

Frame type	0x01
Command Code	0xE1
PL	0x0010
PC + EPC length	0x0E
PC	0x3000
EPC	0x30751FEB705C5904E3D50D70
Parameter	0x00
Checksum	0x3D

Reset ReadProtect:

<< BB 01 E2 00 10 0E 30 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 00 3E 7E

Frame type	0x01
Command Code	0xE2
PL	0x0010
PC + EPC length	0x0E
PC	0x3000
EPC	0x30751FEB705C5904E3D50D70
Parameter	0x00
Checksum	0x3E

If there is no tag response after ReadProtect, an error code of 0x2A will be returned

<< BB 01 FF 00 01 2A 2B 7E

Frame type	0x01
Command Code	0xFF
PL	0x0001
Parameter	0x2A
Checksum	0x2B

If there is no tag response after Reset ReadProtect, an error code of 0x2B will be returned

<< BB 01 FF 00 01 2B 2C 7E

Frame type	0x01
Command Code	0xFF
PL	0x0001
Parameter	0x2A
Checksum	0x2B

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

<< BB 01 FF 00 10 16 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 75 7E

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

3.30 NXP Change EAS (0xE3)

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

Command frame

>> BB 00 E3 00 05 00 00 FF FF 01 E7 7E

Frame type	0x00
Command Code	0xE3
PL	0x0005
Access Password	0x0000FFFF
Set/Reset	0x01 (0x01 sets PSF bit to '1', 0x00 sets PSF bit to '0')
Checksum	0xE7

Response frame

<< BB 01 E3 00 10 0E 30 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 00 3F 7E

Frame type	0x01
Command Code	0xE3
PL	0x0010
PC + EPC length	0x0E
PC	0x3000
EPC	0x30751FEB705C5904E3D50D70
Parameter	0x00
Checksum	0x3F

If there is no tag response, an error code of 0x1B will be returned

<< BB 01 FF 00 01 1B 1C 7E

Frame type	0x01
Command Code	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 and EPC will be sent

<< BB 01 FF 00 10 16 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 75 7E

Frame type	0x01
Command Code	0xFF
PL	0x0010
Error Code	0x16

PC + EPC length	0x0E
PC	0x3400
EPC	0x30751FEB705C5904E3D50D70
Checksum	0x75

3.31 NXP EAS_Alarm (0xE4)

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

Command frame

>> BB 00 E4 00 00 E4 7E

Frame type	0x00
Command Code	0xE4
PL	0x0000
Checksum	0xE4

Response frame

<< BB 01 E4 00 08 69 0A EC 7C D2 15 D8 F9 80 7E

Frame type	0x01
Command Code	0xE4
PL	0x0008
Alarm Code	0x690AEC7CD215D8F9
Checksum	0x80

If there is no tag response, an error code of 0x1D will be returned

<< BB 01 FF 00 01 1D 1E 7E

Frame type	0x01
Command Code	0xFF
PL	0x0001
Parameter	0x1D
Checksum	0x1E

3.32 NXP ChangeConfig (0xE0)

Some series of the NXP G2X tags such as G2iM and G2iM+ support the ChangeConfig command. This command can read or modify the 16-bit Config-Word. The Config-Word is stored at address 0x20 of Bank 01 (EPC MemBank). It can be read or written using the standard Read/Write command. The bits to be toggled in the configuration register need to be set to '1'. Before using this command, a set select parameter command should be sent first to select a single tag.

Command frame

>> BB 00 E0 00 06 00 00 FF FF 00 00 E4 7E

Frame type	0x00
Command Code	0xE0
PL	0x0006
Access Password	0x0000FFFF
Config-Word	0x0000 (all 0 is the same as read, the config word is not changed)
Checksum	0xE4

Response frame

<< BB 01 E0 00 11 0E 30 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 00 41 7E 7E

Frame type	0x01
Command Code	0xE0
PL	0x0011
PC + EPC length	0x0E
PC	0x3000
EPC	0x30751FEB705C5904E3D50D70
Config-Word	0x0041
Checksum	0x7E

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

<< BB 01 FF 00 01 1A 1B 7E

Frame type	0x01
Command Code	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 and EPC will be sent

<< BB 01 FF 00 10 16 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 75 7E

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

3.33 Impinj Monza 4QT (0xE5)

Impinj Monza 4QT tags support the 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 using this command, the set select parameter command should be sent first to select a single tag.

Command frame

Example: set QT_MEM to '1' and write it to Non-volatile memory

>> BB 00 E5 00 08 00 00 FF FF 01 01 40 00 2D 7E

Frame type	0x00
Command Code	0xE5
PL	0x0008
Access Password	0x0000FFFF
Read/Write	0x01 (0x00: Read, 0x01: Write)
Persistence	0x01 (0x00: write to volatile memory, 0x01: write to non-volatile memory, should always be 0x01)
Payload	0x4000 (QT Control, two MSB bits are QT_MEM and QT_SR)
Checksum	0x2D

Response frame

If the operation is successful and the Read/Write field is 0x00, the response frame is:

<< BB 01 E5 00 11 0E 30 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 00 00 42 7E

Frame type	0x01
Command Code	0xE5
PL	0x0011
PC+EPC length	0x0E
PC	0x3000
EPC	0x30751FEB705C5904E3D50D70
QT Control Word	0x0000
Checksum	0x42

If the operation is successful and the Read/Write field is 0x01, the response frame is:

<< BB 01 E6 00 10 0E 30 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 00 42 7E

Frame type	0x01
Command Code	0xE6
PL	0x0010
PC+EPC length	0x0E
PC	0x3000
EPC	0x30751FEB705C5904E3D50D70
Parameter	0x00 (success)
Checksum	0x42

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

<< BB 01 FF 00 01 2E 2F 7E

Frame type	0x01
Command Code	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 and EPC will be sent

<< BB 01 FF 00 10 16 0E 34 00 30 75 1F EB 70 5C 59 04 E3 D5 0D 70 75 7E

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

4 Error Code

If a command execution fails, an error response frame will be returned including the error code to indicate the type of error.

Example: if an inventory command fails due to no tag response or a CRC error, the error code 0x15 will be returned

<< BB 01 FF 00 01 15 16 7E

Frame type	0x01
Command Code	0xFF (indicating error)
PL	0x0001
Parameter	0x15 (error code)
Checksum	0x16

Table of error codes:

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

EPC Gen2 protocol specified error code:

Error-code Support	Error Code	Error code name	Error Description
Error-specific	0b00000000	Other error	Other errors are not described here
	0b00000011	Memory overrun	The given memory area does not exist, or the tag does not support the given EPC length
	0b00000100	Memory locked	The given memory area is locked and/or permanently locked and the lock status is not readable or writable
	0b00001011	Insufficient power	Tag has not enough power to write
Non-specific	0b00001111	Non-specific error	Tag does not support error-code

NXP G2X tags' specific error code:

Type	Code	Description
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 OR Error Code	Specific commands error code. Error code indication refers to the Tag error code in the EPC Gen2 protocol

Impinj Monza QT tags' specific error code:

Type	Code	Description
QT Fail	0x2E	QT failed. No tag response or CRC error
Specific commands error code	0xE0 OR Error Code	Specific commands error code. Error code indication refers to the Tag error code in the EPC Gen2 protocol