

SensArray Communications Protocol: Part II

March 2022





Legal Notices

Copyright ©2018-2022 SensThys, Inc. All rights reserved.

SensThys, Inc and/or its affiliated companies have intellectual property rights relating to technology embodied in the products described in this document, including without limitation certain patents or patent pending applications in the U.S. or other countries.

This document and the products to which it pertains are distributed under licenses restricting their use, copying, distribution and decompilation. No part of this product documentation may be reproduced in any form or by any means without the prior written consent of SensThys, Inc and its suppliers, if any. Third party software is incorporated into this product, and the creators of that software. SensThys, SensArray, SensArray+ and other graphics, logos, and service names used in this document are trademarks of SensThys, Inc and/or its affiliated companies in the U.S. and other countries. All other trademarks are the property of their respective owners. U.S. Government approval required when exporting the product described in this documentation.

Federal Acquisitions: Commercial Software -- Government Users Subject to Standard License Terms and Conditions. U.S. Government: If this Software is being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), then the Government's rights in the Software and accompanying documentation shall be only as set forth in this license; this is in accordance with 48 C.F.R. 227.7201 through 227.7202-4 (for Department of Defense (DoD) acquisitions) and with 48 C.F.R. 2.101 and 12.212 (for non-DoD acquisitions).

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT ARE HEREBY DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

FCC NOTICE: This kit is designed to allow:

(1) Product developers to evaluate electronic components, circuitry, or software associated with the kit to determine whether to incorporate such items in a finished product and

(2) Software developers to write software applications for use with the end product. This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required FCC equipment authorizations are first obtained. Operation is subject to the condition that this product not cause harmful interference to licensed radio stations and that this product accept harmful interference. Unless the assembled kit is designed to operate under part 15, part 18 or part 95 of this chapter, the operator of the kit must operate under the authority of an FCC license holder or must secure an experimental authorization under part 5 of this chapter.

Revision History

Version	Author	Date	Changes
1.0	Brad Gaiser	2018-03-27	Initial Document
1.1	Brad Gaiser	2019-02-12	Added Note for 3.2.39 Set Mask, 3.2.5 Set Gen2 Parameters, and 3.2.37 Set Search Mode.
1.2-1.3			Undocumented changes
1.4	N. Mitchell	2021-11-15	Updated copyright
1.5	N. Mitchell	2022-03-10	Updated copyright



Table of Contents

1	INTRODUCTION	6
1.1	Audience	6
1.2	Type Conventions	6
1.3	System Requirements	7
2	DATA TRANSMISSION FRAME FORMAT	8
2.1	Start of Frame and End of Frame	8
2.2	Length	8
2.3	Commands	8
2.4	Data	8
2.5	CRC	8
3	COMMAND LIST	9
3.1	Version Management	11
3.1.1	Get Hardware Version	11
3.1.2	Get Hardware Version Response	11
3.1.3	Get Firmware Version	11
3.1.4	Get Firmware Version Response	12
3.1.5	Get Reader ID	12
3.1.6	Get Reader ID Response	13
3.2	Reader Control Commands	14
3.2.1	Set Transmit Power	14
3.2.2	Set Transmit Power Response	14
3.2.3	Get Transmit Power	15
3.2.4	Get Transmit Power Response	15
3.2.5	Set Gen2 Parameters	17
3.2.6	Set Gen2 Parameters Response	20
3.2.7	Get Gen2 Parameters	20
3.2.8	Get Gen2 Parameter Response	21
3.2.9	Get Port Return Loss	24
3.2.10	Get Port Return Loss Response	24
3.2.11	Get Temperature	26
3.2.12	Get Temperature Response	26
3.2.13	Set Temperature Protection	26
3.2.14	Set Temperature Protection Response	27
3.2.15	Get Temperature Protection	27
3.2.16	Get Temperature Protection Response	28
3.2.17	Set Inventory Dwell Time	28
3.2.18	Set Inventory Dwell Time response	29
3.2.19	Get Inventory Dwell Time	30
3.2.20	Get Inventory Dwell Time response	30
3.2.21	Get Error Code	30
3.2.22	Get Error Code Response	31
3.2.23	Clear Error Code	31
3.2.24	Clear Error Code Response	32
3.2.25	Set GPIO	33
3.2.26	Set GPIO Response	33
3.2.27	Get GPIO	34
3.2.28	Get GPIO Response	34
3.2.29	Set RF Mode	36
3.2.30	Set RF Mode Response	36
3.2.31	Get RF Mode	36
3.2.32	Get RF Mode Response	37
3.2.33	Get Background RSSI	37
3.2.34	Get Background RSSI Response	38
3.2.35	Software Reset	38
3.2.36	Software Reset Response	39
3.2.37	Set Search Mode	39
3.2.38	Set Search Mode Response	40
3.2.39	Set Mask	41
3.2.40	Set Mask Response	41
3.3	Tag Operation Commands	42
3.3.1	Single Tag Inventory	42

3.3.2	Single Tag Inventory Response	42
3.3.3	Start Inventory	44
3.3.4	Start Inventory Response	44
3.3.5	Stop Inventory	45
3.3.6	Stop Inventory Response	45
3.3.7	Read Tag	47
3.3.8	Read Tag Response	47
3.3.9	Write Tag	48
3.3.10	Write Tag Response	49
3.3.11	Lock Tag	50
3.3.12	Lock Tag Response	50
3.3.13	Kill Tag	52
3.3.14	Kill Tag Response	52
3.3.15	Timed Inventory	52
3.3.16	Timed Inventory Response	53
3.3.17	Get Timed Inventory Result	53
3.3.18	Block Write	54
3.3.19	Block Write Response	55
3.3.20	Block Erase Tag	56
3.3.21	Block Erase Tag Response	56
3.3.22	Operation Failed Response Frame	56

1 Introduction

This document describes the various commands in the low-level binary protocol used to communicate with the reader module contained within the SensArray Integrated RFID Reader. There are three sets of commands that are part of the communication protocol:

1. Networking and configuration commands that are handled by the SensArray reader's command processor.
2. Commands that are handled by the embedded RFID reader module.
3. Commands where the underlying functionality is handled by the embedded RFID module but where the command processor provides some additional oversight and extended functionality.

A more detailed description of each of these three categories is provided in Part I of this protocol guide.

This document covers the second topic, outlining the communication protocol for the reader module within the SensArray and SensArray+. This protocol enables reader control and data transfer to a host computer communicating is through Ethernet TCP socket 5000 to either a SensArray or SensArray+ integrated reader.

Part I discusses the commands that implement networking configuration and control as well as commands that are part of the underlying reader module protocol, but which the main SensArray microcontroller intervenes to provide extended capabilities.

1.1 Audience

We assume that the readers of this guide:

- Understand the fundamentals of binary protocols,
- Are familiar with serial communication interfaces, and
- Have some previous knowledge of RFID, and other relevant technologies

1.2 Type Conventions

Regular text appears in a plain, Arial font:

Regular text format

Example command sequences appear in a fixed, Courier font:

Example Command format

1.3 System Requirements

Any computer or embedded controller that can be networked to the reader.

2 Data Transmission Frame Format

Commands and data are transmitted in “frames”. A data frame consists of a Start of Frame (SOF), Length, Command, Data, CRC and End Of Frame (EOF), as shown below:

SOF	Length	Command	Data	CRC	EOF
2 Bytes	2 Bytes	1 Byte	N Bytes	1 Byte	2 Bytes

2.1 Start of Frame and End of Frame

The Start Of Frame (SOF) sequence denotes the start of the data frame. SOF is 2 bytes: **0xA5 0x5A**.

The End Of Frame (EOF) sequence denotes the end of the data frame. EOF is 2 bytes: **0x0D 0x0A**.

2.2 Length

The Length field represents the length of the complete data transmission frame, calculated as follows:

Length = 2+2+1+N+1+2, where N is the length of the Data field.

2.3 Commands

The supported commands are shown in the table below.

2.4 Data

The “Data” field contains either the command parameters (“Command” frame) or the results from a reader operation (“Response” frame).

2.5 CRC

CRC value is calculated using bitwise XOR of the Data bytes in each frame.

Example:

FRAME = 0xA5 0x5A 0x00 0x0A 0x43 0x01 0x25 CRC 0x0D 0x0A

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

3 Command List

Description	Command
Get Hardware Version	0x00
Get Hardware Version response	0x01
Get Firmware Version	0x02
Get Firmware Version response	0x03
Get Reader ID	0x04
Get Reader ID response	0x05
Set Transmit Power	0x10
Set Transmit Power response	0x11
Get Transmit Power	0x12
Get Transmit Power response	0x13
Set Gen2 Parameters	0x20
Set Gen2 Parameters response	0x21
Get Gen2 Parameters	0x22
Get Gen2 Parameters response	0x23
Get Port Return Loss	0x32
Get Port Return Loss response	0x33
Get Temperature	0x34
Get Temperature response	0x35
Set Temperature Protection	0x38
Set Temperature Protection response	0x39
Get Temperature Protection	0x3A
Get Temperature Protection response	0x3B
Set Inventory Dwell Time	0x3C
Set Inventory Dwell Time response	0x3D
Get Inventory Dwell Time	0x3E
Get Inventory Dwell Time response	0x3F
Get Error Flag	0x40
Get Error Flag response	0x41
Clear Error	0x42
Clear Error response	0x43
Set GPIO	0x46
Set GPIO response	0x47
Get GPIO	0x48
Get GPIO response	0x49
Set RF Mode	0x52
Set RF Mode response	0x53

Description	Command
Get RF Mode	0x54
Get RF Mode response	0x55
Get Background RSSI Value	0x64
Get Background RSSI Value response	0x65
Set Baud Rate	0x66
Set Baud Rate response	0x67
Software Reset	0x68
Software Reset response	0x69
Set Search Mode	0x6A
Set Search Mode response	0x6B
Set Mask	0x6E
Set Mask response	0x6F
Single Tag Inventory	0x80
Single Tag Inventory response	0x81
Start Inventory	0x82
Start Inventory response	0x83
Stop Inventory	0x8c
Stop Inventory response	0x8d
Read Tag	0x84
Read Tag response	0x85
Write Tag	0x86
Write Tag response	0x87
Lock Tag	0x88
Lock Tag response	0x89
Kill tag	0x8a
Kill tag response	0x8b
Timed Inventory	0x90
Timed Inventory response	0x91
Get Timed Inventory result	0x92
Block Write tag	0x93
Block Write tag response	0x94
Block Erase tag	0x95
Block Erase tag response	0x96
Operation Failed response	0xff

3.1 Version Management

3.1.1 Get Hardware Version

Length: 0x00 0x08

Command: 0x00

Data: none

CRC: 0x08

Function: Get hardware version

Get Hardware Version command frame

SOF	Length	Cmd	Data	CRC	EOF
0xA5 0x5A	0x00 0x08	0x00	None	0x08	0x0D 0x0A

Example: Get Hardware Version

Command: A5 5A 00 08 00 08 0D 0A

3.1.2 Get Hardware Version Response

Length: 0x00 0x0B

Command: 0x01

Data: 3 bytes: Major Version, Minor Version and Revision.

Function: Response to the Get Hardware Version command

Get Hardware Version response frame

SOF	Length	Cmd	Data	CRC	EOF
0xA5 0x5A	0x00 0x0B	0x01	Major Ver Minor Ver Revision	0xXX	0x0D 0x0A

Example: Response with hardware version 2.3.5

Command: A5 5A 00 0B 01 02 03 05 0E 0D 0A

3.1.3 Get Firmware Version

Length: 0x00 0x08

Command: 0x02
Data: None
CRC: 0x0A

Function: Get firmware version

Get Firmware Version command frame

SOF	Length	Cmd	Data	CRC	EOF	EOF
0xA5	0x5A	0x00	0x08	0x02	none	0x0A

Example: Get Firmware Version

Command: A5 5A 00 08 02 0A 0D 0A

3.1.4 Get Firmware Version Response

Length: 0x00 0x0B
Command: 0x03
Data: 3 bytes: Major Version, Minor Version, Revision

Function: Reply to the Get Firmware Version command

Get Hardware Version response frame

SOF	Length	Cmd	Data	CRC	EOF
0xA5	0x5A	0x00	0x0B	0x03	Major Ver Minor Ver Revision
				0xXX	0x0D 0x0A

Example: Response with firmware version 2.3.5

Command: A5 5A 00 0B 03 02 03 05 0C 0D 0A

3.1.5 Get Reader ID

Length: 0x00 0x08
Command: 0x04
Data: None
CRC: 0x0C

Function: Get Reader ID

Get Reader ID command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x04	None	0x0C	0x0D	0x0A

Example: Get Reader ID

Command: A5 5A 00 08 04 0C 0D 0A

3.1.6 Get Reader ID Response

Length: 0x00 0x0C

Command: 0x05

Data: 4 bytes of unique Reader ID

Function: The response to the Get Reader ID command.

Get Reader ID response frame

SOF		Length		Cmd	Data				CRC	EOF	
0xA5	0x5A	0x00	0x0C	0x05	DByte3	DByte2	DByte1	DByte0	0xxx	0x0D	0x0A

Example: Unique Reader ID is 0x00000530 (1328)

Command: A5 5A 00 0C 05 00 00 05 30 3C 0D 0A

3.2 Reader Control Commands

3.2.1 Set Transmit Power

Length: 0x00 0x0E
 Command: 0x10
 Data: 6 bytes: Mode, Antenna ID,
 Read Power (2 bytes, dBm)
 Write Power (2 bytes, dBm)

Function: Set antenna read/write power.

Set Transmit Power command frame

SOF		Length		Cmd	Data				CRC	EOF	
0xA5	0x5A	0x00	0x0E	0x10	Mode	Ant	2:ReadPwr	2:WritePwr	0xxx	0x0D	0x0A

Bit7 – Bit2	Bit1	Bit0
RFU	0x00=don't save 0x01=save	0x00=open loop 0x01=closed loop

The Mode bits

Mode:

bit7-bit2 are reserved
 bit0: 0x00=open loop (default), 0x01=closed loop
 (NOTE: closed loop mode is not currently supported)
 bit1: save the value to NVM (0x01=save, 0x00=do not save)

Antenna ID is a 0 based antenna number.

Read/Write Power data bytes are the power value in dBm x100 (signed 16-bit value)

Example: Set antenna 1 read power 15dBm, write power 30dBm, open loop, not saved
 Command: A5 5A 00 0E 10 00 01 05 DC 0B B8 77 0D 0A

3.2.2 Set Transmit Power Response

Length: 0x00 0x09
 Command: 0x11

Data: Result (0x01=OK, 0x00=failed)

Function: Response to the Set Transmit Power command

Set Transmit Power response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x11	0x01=OK 0x00=fail	0xFF	0x0D	0x0A

Example: Set transmit power succeeded

Command: A5 5A 00 09 11 01 19 0D 0A

3.2.3 Get Transmit Power

Length: 0x00 0x08

Command: 0x12

Data: None

Function: Get Transmit Power

Get Transmit Power command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x12	None	1A	0x0D	0x0A

Example: Get Transmit Power

Command: A5 5A 00 08 12 1A 0D 0A

3.2.4 Get Transmit Power Response

Length: variable

Command: 0x13

Data: Mode, [Antenna ID, read power (2 bytes, dBm), write power (2 bytes, dBm)]

Function: Get read/write power for each antenna

Get Transmit Power response frame

SOF		Length		Cmd		Data	
0xA5	0x5A	0xFF	0xFF	0x13	Mode	Ant ID	R(MSB)
Data							
R(LSB)	W(MSB)	W(LSB)	Ant ID	R(MSB)	R(LSB)	W(MSB)	W(LSB)
Data				CRC		EOF	
...	Ant ID	R(MSB)	R(LSB)	W(MSB)	W(LSB)	0xxx	0x0D 0x0A

Mode (0x00=open loop, 0x01=closed loop)

Read/Write Power data bytes are the power value in dBm x 100 (signed 16-bit value).

Example: Closed loop mode, antenna 1 has read power -7dBm, write power 21dBm,
antenna 1 has read power 3dBm, write power 14dBm.

Command: A5 5A 00 13 13 01 01 FD 44 08 34 01 01 2C 05 78 D0 0D 0A

3.2.5 Set Gen2 Parameters

Length: 0x00 0x10

Command: 0x20

Data: see below

Function: Set Gen2 inventory parameters.

Set Gen2 Parameters command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x10	0x20	DByte7 – Dbyte0	0xXX	0x0D	0x0A

Data definition

DByte7									DByte6									DByte5													DByte4							
Target			Action			Tr	Q		StartQ				MinQ					MaxQ				D	E	P		Sel		Ses	T	LF								
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x							
DByte3									DByte2									DByte1													Dbyte0							
RFU																																						
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							

Target (inventory target)

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

Action (inventory action)

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

(Tr) Truncate

Disable truncation	B'0
Enable truncation	B'1

Q setting:

Fixed Q	B'0
Dynamic Q	B'1

Note: in the Fixed Q mode, the Q is fixed as StartQ. The values of MinQ and MaxQ are ignored.

StartQ setting:

0 - 15	B'0000 - B'1111
--------	-----------------

MinQ setting:

0 - 15	B'0000 - B'1111
--------	-----------------

MaxQ setting:

0 - 15	B'0000 - B'1111
--------	-----------------

(D) DR parameter values

8	B'0
64/3	B'1

(E) Encoding parameter values

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

(P) TRext (pilot tone)

No pilot tone	B'0
Use pilot tone	B'1

Sel parameter values

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

Session

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

(T) Target

A	B'0
B	B'1

(LF) Link Frequency

40KHz	B'000
160KHz	B'001
200KHz	B'010
250KHz	B'011
300KHz	B'100
320KHz	B'101
400KHz	B'110
640KHz	B'111

Example: Target is S0, Action is B'100, Truncate is Disable Truncate, use dynamic Q mode, startQ is 4, minQ is 0, maxQ is 15, Dr is 8, M is Miller4, TRext is No pilot tone, Sel is ALL, Session is S2, Target is A, Link Frequency is 160KHz.

Command: A5 5A 00 10 20 11 40 F4 21 00 00 00 00 B4 0D 0A

NOTE: Perform Set Mask first (see 3.2.39) then Set Gen2 Parameters to ensure that parameters are properly masked.

3.2.6 Set Gen2 Parameters Response

Length: 0x00 0x09
 Command: 0x21
 Data: Result (0x01=OK, 0x00=failed)

Function: Set Gen2 Parameters response

Set Gen2 Parameters response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x21	0x01=OK 0x00=failed	0xFF	0x0D	0x0A

Example: Set Gen2 parameters succeeded.

Command: A5 5A 00 09 21 01 29 0D 0A

3.2.7 Get Gen2 Parameters

Length: 0x00 0x08
 Command: 0x22
 Data: None

Function: Get Gen2 inventory parameters.

Get Gen2 Parameters command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x22	None	0x2A	0x0D	0x0A

Example: Get Gen2 parameters

Command: A5 5A 00 08 22 2A 0D 0A

3.2.8 Get Gen2 Parameter Response

Length: 0x00 0x10
 Command: 0x23
 Data: Gen2 inventory parameter values

Function: Get Gen2 Parameters response

Get Gen2 Parameters response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x10	0x23	DByte7 – Dbyte0	0xXX	0x0D	0x0A

Data definition

DByte7									DByte6								DByte5										DByte4							
Target			Action			Tr	Q	StartQ				MinQ				MaxQ			D	E	P	Sel		Ses		T	LF							
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x					
DByte3									DByte2								DByte1										Dbyte0							
RFU																																		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					

Target (inventory target)

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

Action (inventory action)

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

(Tr) Truncate

Disable truncation	B'0
Enable truncation	B'1

Q setting:

Fixed Q	B'0
Dynamic Q	B'1

Note: when fixed Q mode is set, the Q is fixed as StartQ. The values of MinQ and MaxQ are ignored.

StartQ setting:

0 - 15	B'0000 - B'1111
--------	-----------------

MinQ setting:

0 - 15	B'0000 - B'1111
--------	-----------------

MaxQ setting:

0 - 15	B'0000 - B'1111
--------	-----------------

(D) DR parameter values

8	B'0
64/3	B'1

(E) Encoding parameter values

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

(P) TRext (Pilot tone)

No pilot tone	B'0
Use pilot tone	B'1

Sel parameter values

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

Session

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

(T) Target

A	B'0
B	B'1

(LF) Link Frequency

40KHz	B'000
160KHz	B'001
200KHz	B'010
250KHz	B'011
300KHz	B'100
320KHz	B'101
400KHz	B'110
640KHz	B'111

Example:

Target is S0, Action is B'100, Truncate data is Disable Truncate, use Dynamic Q mode, startQ is 4, minQ is 0, maxQ is 15, DR is 8, M data is Miller4, Trest data is No pilot tone, Sel is ALL, Session is S2, Target is A, Link Frequency is 160KHz

Command: A5 5A 00 10 23 11 40 F4 21 00 00 00 00 B7 0D 0A

3.2.9 Get Port Return Loss

Length: 0x00 0x08

Command: 0x32

Data: None

Function: Get current reflected power

Get Port Return Loss command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x32	None	0x3A	0x0D	0x0A

Example: Get Port Return Loss

Command: A5 5A 00 08 32 3A 0D 0A

3.2.10 Get Port Return Loss Response

Length: 0x00 0x0B

Command: 0x33

Data: Result (0x01=success, 0x00=failed), loss (2 bytes)

Function: Get Port Return Loss response

Get Port Return Loss response frame

SOF		Length		Cmd	Data			CRC	EOF	
0xA5	0x5A	0x00	0x0B	0x33	0x01=OK 0x00=fail	Loss (MSB)	Loss (LSB)	0xFF	0x0D	0x0A

Return Loss data conversion

0x0000	-25dBm	0x0126	-14dBm	0x0BFF	-3dBm	0x46FF	8dBm
0x0000	-24dBm	0x0131	-13dBm	0x0EFF	-2dBm	0x50FF	9dBm
0x0000	-23dBm	0x013E	-12dBm	0x10FF	-1dBm	0x5DFF	10dBm
0x0001	-22dBm	0x024C	-11dBm	0x14FF	0dBm	0x6AFF	11dBm
0x0002	-21dBm	0x0260	-10dBm	0x17FF	1dBm	0x7AFF	12dBm
0x0005	-20dBm	0x0374	-9dBm	0x1CFF	2dBm	0x8BFF	13dBm
0x0007	-19dBm	0x048B	-8dBm	0x21FF	3dBm	0x9EFF	14dBm
0x000B	-18dBm	0x05A5	-7dBm	0x26FF	4dBm	0xB4FF	15dBm
0x0010	-17dBm	0x06C3	-6dBm	0x2DFF	5dBm	0xCCFF	16dBm
0x0116	-16dBm	0x08E6	-5dBm	0x34FF	6dBm	0xE7FF	17dBm
0x011D	-15dBm	0x09FF	-4dBm	0x3CFF	7dBm	0xFFFF	18dBm

Example: Get succeeded, return loss = -3dBm

Command: A5 5A 00 0B 33 0x01 26 18 07 0D 0A

Note: -3 dBm is from the table as 0x2618 is between 0x21FF and 0x26FF

3.2.11 Get Temperature

Length: 0x00 0x08
 Command: 0x34
 Data: None

Function: Get current module temperature ($\pm 6^{\circ}\text{C}$).

Get Temperature command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x34	None	0x3C	0x0D	0x0A

Example: Get Temperature

Command: A5 5A 00 08 34 3C 0D 0A

3.2.12 Get Temperature Response

Length: 0x00 0x0B
 Command: 0x35
 Data: Result (0x01=OK, 0x00=failed),
 Temperature (2 bytes, $\times 100^{\circ}\text{C}$)

Function: Get Temperature response

Get Temperature response frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0B	0x35	0x01=OK 0x00=failed	Temp (MSB) Temp (LSB)	0xXX	0x0D	0x0A

The temperature in $^{\circ}\text{C} \times 100$ (signed 16-bit number)

Example: Command succeeded, temperature is -40°C

Command: A5 5A 00 0B 35 01 F0 60 AF 0D 0A

3.2.13 Set Temperature Protection

Length: 0x00 0x09
 Command: 0x38

Data: 1 byte, temperature protection (0x00=OFF, 0x01=ON)

Function: Set Temperature Protection.

Set Temperature Protection command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x38	Data0	0xFF	0x0D	0x0A

When the temperature protection is enabled, and the temperature exceeds 65°C, the rate of inventory reads will automatically decrease by 50%. If the temperature exceeds 85°C, the module will automatically shut down. The module will automatically restart after the temperature falls below 85°C. Default value is ON.

Note: Disabling temperature protection is not recommended and voids warranty.

Example: Set Temperature Protection

Command: A5 5A 00 09 38 01 30 0D 0A

3.2.14 Set Temperature Protection Response

Length: 0x00 0x09

Command: 0x39

Data: Result (0x01=OK, 0x00=failed)

Function: Set Temperature Protection response

Set Temperature Protection response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x39	0x01=OK 0x00=fail	0xFF	0x0D	0x0A

Example: Set succeeded

Command: A5 5A 00 09 39 01 31 0D 0A

3.2.15 Get Temperature Protection

Length: 0x00 0x08

Command: 0x3A

Data: None

CRC: 0x32

Function: Get Temperature Protection Setting.

Get Temperature Protection Setting command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x3A	None	0x32	0x0D	0x0A

When the temperature protection is enabled, and the temperature exceeds 65°C, the rate of inventory reads will automatically decrease by 50%. If the temperature exceeds 85°C, the module will automatically shut down. The module will automatically restart after the temperature falls below 85°C. Default value is ON.

Example: Get the Temperature Protection setting

Command: A5 5A 00 08 3A 32 0D 0

3.2.16 Get Temperature Protection Response

Length: 0x00 0x0A
 Command: 0x3B
 Data: Result (0x01=OK, 0x00=failed)

Function: Get the Temperature Protection response.

Get Temperature Protection response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x3B	0x01=OK, 0x00=failed	Data0	0xXX	0x0D 0x0A

Data0: Temperature protection (0x00=ON, 0x01=OFF)

Example: Command succeeded, temperature protection is set to ON.

Command: A5 5A 00 0A 3B 01 01 31 0D 0A

3.2.17 Set Inventory Dwell Time

Length: 0x00 0x0D
 Command: 0x3C
 Data: 5 bytes

Function: Set inventory dwell (work and wait) time

Set Inventory Dwell Time frame

SOF		Length		Cmd	Data					CRC	EOF	
0xA5	0x5A	0x00	0x0D	0x3C	Save	Work MSB	Work LSB	Wait MSB	Wait LSB	0xxx	0x0D	0x0A

The settings will be saved to NVM if 'Save' is 1.

Work and Wait times are in milliseconds, 16-bit numbers (0-65535)

Example: Set 50 msec Work time and 100 msec wait time, save to NVM.

Command: A5 5A 00 0D 3C 01 00 32 00 64 66 0D 0A

3.2.18 Set Inventory Dwell Time response

Length: 0x00 0x09

Command: 0x3D

Data: Result (0x01=OK, 0x00=failed)

Function: Response to the Set Inventory Dwell time command

Set Inventory Dwell Time response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x3C	0x01=OK, 0x00=failed	0xXX	0x0D	0x0A

Example: Command succeeded.

Command: A5 5A 00 09 3D 01 35 0D 0A

3.2.19 Get Inventory Dwell Time

Length: 0x00 0x08
 Command: 0x3E
 Data: none

Function: Get inventory dwell (work and wait) time

Get Inventory Dwell Time frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x3E	none	0xFF	0x0D	0x0A

Example: Get inventory dwell time.

Command: A5 5A 00 08 3E 36 0D 0A

3.2.20 Get Inventory Dwell Time response

Length: 0x00 0x0D
 Command: 0x3F
 Data: Result (0x01=OK, 0x00=failed)

Function: Response to the Get Inventory Dwell time command

Get Inventory Dwell Time response frame

SOF		Length		Cmd	Data					CRC	EOF	
0xA5	0x5A	0x00	0x0D	0x3F	0x01=OK 0x00=failed	Work MSB	Work LSB	Wait MSB	Wait LSB	0xFF	0x0D	0x0A

Work and Wait times are in milliseconds, 16-bit numbers (0-65535)

Example: Command succeeded. Work time is 50 msec, Wait time is 100 msec

Command: A5 5A 00 0D 3F 01 00 32 00 64 65 0D 0A

3.2.21 Get Error Code

Length: 0x00 0x08
 Command: 0x40
 Data: None

Function: Get error code

Get Error Code command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x40	None	0x48	0x0D	0x0A

Example: Get Error code

Command: A5 5A 00 08 40 48 0D 0A

3.2.22 Get Error Code Response

Length: 0x00 0x0B
 Command: 0x41
 Data: Result (0x01=OK, 0x00=failed)

Function: Get error code response

Get Error Code response frame

SOF		Length		Cmd	Data			CRC	EOF	
0xA5	0x5A	0x00	0x0B	0x41	0x01=OK 0x00=failed	Error Code (MSB)	Error Code (LSB)	0xXX	0x0D	0x0A

Example: Get Error Code command succeeded, the Error Code is 0x0402

Command: A5 5A 00 0B 41 01 04 02 4D 0D 0A

3.2.23 Clear Error Code

Length: 0x00 0x08

Command: 0x42
Data: None

Function: Clear error code

Clear Error Code command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x42	None	0x4A	0x0D	0x0A

Example: Clear error code

Command: A5 5A 00 08 42 4A 0D 0A

3.2.24 Clear Error Code Response

Length: 0x00 0x09
Command: 0x43
Data: Result (0x01=OK, 0x00=failed)

Function: Clear Error Code response

Clear Error Code response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x43	0x01=OK 0x00=fail	0xXX	0x0D	0x0A

Example: Clear error code succeeded

Command: A5 5A 00 09 43 01 4B 0D 0A

3.2.25 Set GPIO

Length: 0x00 0x0A
 Command: 0x46
 Data: 2 bytes

Function: Set GPIO

Set GPIO command frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x46	DByte1	DByte0	0xxx	0x0D	0x0A

GPIO data definition

DByte1							
GPIO8	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1
DByte0							
H/L	H/L	H/L	H/L	H/L	H/L	H/L	H/L

Dbyte1 is the bitmap of GPIO ports to set. DByte0 is the GPIO ports output state: 1=high, 0=low.

The number of GPIO ports is hardware dependent, GPIO1-GPIO4 are outputs, GPIO5-GPIO8 are inputs.

Example: GPIO1 is High and GPIO3 is low

Command: A5 5A 00 0A 46 05 01 4A 0D 0A

3.2.26 Set GPIO Response

Length: 0x00 0x09
 Command: 0x47
 Data: Result (0x01=OK, 0x00=failed)

Function: Set GPIO response

Set GPIO response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x47	0x01=OK 0x00=fail	0xXX	0x0D	0x0A

Example: Set GPIO succeeded

Command: A5 5A 00 09 47 01 4E 0D 0A

3.2.27 Get GPIO

Length: 0x00 0x0A

Command: 0x48

Data: 2 bytes

Function: Get GPIO state

Get GPIO state command frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x48	DByte0	RFU	0xFF	0x0D	0x0A

GPIO DByte0 definition

DByte0							
GPIO8	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1

DByte0 is the bitmap of GPIO ports to fetch.

Example: Get GPIO3 state

Command: A5 5A 00 0A 48 04 00 46 0D 0A

3.2.28 Get GPIO Response

Length: 0x00 0x0B

Command: 0x49

Data: 3 bytes.

Function: Get GPIO response

Get GPIO response frame

SOF		Length		Cmd	Data			CRC	EOF	
0xA5	0x5A	0x00	0x0B	0x49	0x01=OK 0x00=fail	DByte1	DByte0	0xxx	0x0D	0x0A

Data definition

DByte1							
GPIO8	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1
DByte0							
H/L	H/L	H/L	H/L	H/L	H/L	H/L	H/L

DByte1 is the bitmap of GPIO ports (1=port is present).

DByte2 is the port state (1=output is HIGH, 0=GPIO output is low).

Example: response to the command to fetch the states of GPIO2 and GPIO3: GPIO3 is High and GPIO1 is Low

Command: A5 5A 00 0B 49 01 05 04 43 0D 0A

3.2.29 Set RF Mode

Length: 0x00 0x0B
 Command: 0x52
 Data: 3 bytes
 Function: Set RF mode

Set RF Mode command frame

SOF		Length		Cmd	Data			CRC	EOF	
0xA5	0x5A	0x00	0x0B	0x52	RFU	DByte1	DByte0	0xXX	0x0D	0x0A

The RF mode value will be saved to NVM if DByte1 is 1. The following RF modes are supported:

DByte0	RF Mode
0x00	DSB_ASK /FM0/ 40 KHz
0x01	PR_ASK /Miller4/ 250KHz
0x02	PR_ASK /Miller4/ 300KHz
0x03	DSB_ASK /FM0/ 400KHz

Example: Set the RF mode to DSB_ASK /FM0/ 40 KHz, without saving to NVM.

Command: A5 5A 00 0B 52 00 00 00 59 0D 0A

3.2.30 Set RF Mode Response

Length: 0x00 0x09
 Command: 0x53
 Data: Result (0x01=OK, 0x00=failed)
 Function: Set RF Mode response

Set RF Mode response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x53	0x01=OK 0x00=fail	0xXX	0x0D	0x0A

Example: Set RF Mode succeeded

Command: A5 5A 00 09 53 01 5B 0D 0A

3.2.31 Get RF Mode

Length: 0x00 0x0A
 Command: 0x54
 Data: 2 bytes
 Function: Get RF mode.

Get RF Mode command frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x54	RFU	RFU	0xXX	0x0D	0x0A

Example: Get RF mode

Command: A5 5A 00 0A 54 00 00 5E 0D 0A

3.2.32 Get RF Mode Response

Length: 0x00 0x0B
 Command: 0x55
 Data: 3 bytes
 Function: Get RF Mode response frame.

Get RF Mode response frame

SOF		Length		Cmd	Data			CRC	EOF	
0xA5	0x5A	0x00	0x0B	0x55	0x01=OK 0x00=fail	RFU	DByte0	0xXX	0x0D	0x0A

Data definition

DByte0	RF link
0x00	DSB_ASK /FM0/ 40 KHz
0x01	PR_ASK /Miller4/ 250 KHz
0x02	PR_ASK /Miller4/ 300 KHz
0x03	DSB_ASK /FM0/ 400 KHz

Example: Current RF mode is DSB_ASK / FM0 / 400 KHz.

Command: A5 5A 00 0B 55 01 00 03 58 0D 0A

3.2.33 Get Background RSSI

Length: 0x00 0x0A
 Command: 0x64

Data: None

Function: Get Background RSSI.

Get Background RSSI command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x64	None	0x6C	0x0D	0x0A

Example: Get Background RSSI

Command: A5 5A 00 08 64 6C 0D 0A

3.2.34 Get Background RSSI Response

Length: 0x00 0x0B

Command: 0x65

Data: 3 bytes

Function: Get Background RSSI response.

Get Background RSSI value response frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0B	0x65	0x01=OK 0x00=fail	Data1 Data0	0xXX	0x0D	0x0A

RSSI (dBm) is the data value multiplied by 10 (signed 2 byte number).
For example, data=FD 6F, RSSI=-65.7dBm.

Example: Command succeeded. RSSI value is -65.7dBm.

Command: A5 5A 00 0B 65 01 FD 6F FD 0D 0A

3.2.35 Software Reset

Length: 0x00 0x08

Command: 0x68

Data: None

Function: Software Reset

Software Reset command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x68	None	0x60	0x0D	0x0A

Example: Send software reset command

Command: A5 5A 00 08 68 60 0D 0A

3.2.36 Software Reset Response

Length: 0x00 0x09
 Command: 0x69
 Data: 1 byte: Result (0x01=OK, 0x00=failed)

Function: Software Reset response

Software Reset response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x69	0x01=OK, x00=fail	0xXX	0x0D	0x0A

Example: Reset succeeded

Command: A5 5A 00 09 69 01 61 0D 0A

3.2.37 Set Search Mode

Length: 0x00 0x0A
 Command: 0x6A
 Data: 2 bytes

Function: Set inventory Search Mode (dual/single target)

Set Search Mode frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x6A	DByte1	DByte0	0xFF	0x0D	0x0A

DByte1 controls whether to save the Search Mode value to NVM: 0x01=save, 0x00=do not save.

DByte0 is the search mode (inventory target): 0x00=dual (default), 0x01=single inventory target

Example: Set single target search and save to NVM

Command: A5 5A 00 0A 6A 01 01 60 0D 0A

NOTE: Perform Set Mask first (see 3.2.39) then Set Gen2 Parameters to ensure that parameters are properly masked. Followed by Set Search Mode.

3.2.38 Set Search Mode Response

Length: 0x00 0x09

Command: 0x6B

Data: 1 byte: Result (0x01=OK, 0x00=failed)

Function: Set inventory Search Mode (dual/single target) response

Set Search Mode response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x6B	0x01=OK, 0x00=fail	0xFF	0x0D	0x0A

Example: Command succeeded

Command: A5 5A 00 09 6B 01 63 0D 0A

3.2.39 Set Mask

Length: variable
 Command: 0x6E
 Data: mask data bytes
 Function: Set inventory mask
 Set Mask command frame

SOF		Length		Cmd	Data			
0xA5	0x5A	0x00	0xFF	0x6E	DByte0	Bank	MaskAddr (MSB)	MaskAddr (LSB)
Data					CRC		EOF	
MaskLen (MSB)	MaskLen (LSB)	MaskData (MSB)	...	MaskData (LSB)	0xFF	0x0D	0x0A	

DByte0 controls whether to save the Mask value to NVM (0x01=save, 0x00=do not save)

Mask Bank (0x01=EPC, 0x02=TID, 0x03=USER), Mask Address in bits, Mask Length in bits (0x00=no mask), Mask Data bytes (right-padded with zeroes if Mask Length is not byte aligned)

Example 1: Set inventory mask on TID bank at address of 0, length is 96 bits, mask data is 0xE2003414013301001038D2B5, save to NVM.

Command: A5 5A 00 1A 6E 01 02 00 00 00 60 E2 00 34 14 01 33 01 00 10 38 D2 B5 A9 0D 0A

Example 2: Clear inventory mask, save to NVM

Command: A5 5A 00 0E 6E 01 00 00 00 00 00 61 0D 0A

NOTE: Perform Set Mask first (see 3.2.39) then Set Gen2 Parameters to ensure that parameters are properly masked.

3.2.40 Set Mask Response

Length: 0x00 0x09
 Command: 0x6F
 Data: 1 byte: Result (0x01=OK, 0x00=failed)
 Function: Set inventory mask response
 Set Mask response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x6F	0x01=OK 0x00=failed	0xFF	0x0D	0x0A

Example: Set mask succeeded

Command: A5 5A 00 09 6F 01 67 0D 0A

3.3 Tag Operation Commands

3.3.1 Single Tag Inventory

Length: 0x00 0x0A
 Command: 0x80
 Data: 2 bytes: the inventory timeout value in msec. The inventory stops as soon as either a tag is found or the timeout occurs.

Function: Perform a single inventory

Single Tag Inventory command frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x80	Timeout (MSB)	Timeout (LSB)	0xxx	0x0D	0x0A

Example: Perform a Single Tag inventory, timeout is 100ms (0x0064).

Command: A5 5A 00 0A 80 00 64 EE 0D 0A

3.3.2 Single Tag Inventory Response

Length: Variable
 Command: 0x81
 Data: PC+EPC, RSSI, Antenna number

Function: Single Tag Inventory response.

Single Tag Inventory response frame

SOF		Length		Cmd	Data								CRC	EOF	
0xA5	0x5A	0x00	0xXX	0x81	PC (MSB)	PC (LSB)	EPC (MSB)	...	EPC (LSB)	RSSI (MSB)	RSSI (LSB)	Ant	0xXX	0x0D	0x0A

RSSI is reported as a 16-bit signed value. Units are x10 dBm.

For example, RSSI=FD6F represents the RSSI value of -65.7dBm.

NOTE: The frame length is variable and depends on the EPC length

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

Command:

```
A5 5A 00 19 81 30 00 E2 00 34 11 B8 02 01 13 83 25 85 66 FD
6F 02 12 0D 0A
```

3.3.3 Start Inventory

Length: 0x00 0x0A
 Command: 0x82
 Data: 2 bytes: the number of inventory cycles
 Function: Perform multiple inventory cycles.

Start Inventory command frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x82	Num[1]	Num[0]	0xxx	0x0D	0x0A

The Data (0x0000-0xFFFF) is the number of inventory cycles to perform. Setting the number of inventory cycles to 0 will make it repeat inventories indefinitely.

NOTE: The reader will not respond to any other commands while performing inventories. To stop performing inventories, send the Stop Inventory command.

Example: Perform 10000 inventory cycles (0x2710).

Command: A5 5A 00 0A 82 27 10 BF 0D 0A

3.3.4 Start Inventory Response

Length: Variable
 Command: 0x83
 Data: PC+EPC, RSSI, Antenna number, Reader time in seconds since 1/1/1970 at 0:00:00 (epoch time) + 1/100ths of seconds, AFEC bits.

Function: Response to the Start Inventory command

Start Inventory Response frame

SOF		Length		Cmd	Data					
0xA5	0x5A	0x00	0xXX	0x83	PC (MSB)	PC (LSB)	EPC (MSB)	...	EPC (LSB)	
									CRC	EOF
RSSI (MSB)	RSSI (LSB)	Ant	Epoch Time (Byte 0)	...	Epoch Time (Byte 3)	Epoch Time 100ths sec	AFEC Info (1 Byte)	0xXX	0x0D	0x0A

RSSI is reported as a 16-bit signed value. Units are x10 dBm.

For example, RSSI=FD6F represents the RSSI value of -65.7dBm.

The AFEC bits are as follows:

- LSB bit (bit 0): CRC error = 1, no error = 0

- Bit 1: FEC calculation error = 1, no error = 0
- Bit 2: Tag data unrecoverable = 1, recoverable/valid = 0

A tag read with no errors will return 0 for this byte. Recoverable data will have bit 2 set to 0, but underlying errors will be indicated by bits 0 and 1.

NOTE: The frame length is variable and depends on the EPC length.

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

Command: A5 5A 00 19 83 30 00 E2 00 34 11 B8 02 01 13 83 25 85 66 FD 6F 02 10 0D 0A

3.3.5 Stop Inventory

Length: 0x00 0x08
 Command: 0x8C
 Data: None

Function: Stop performing repeated inventory cycles

Stop Inventory command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x8C	None	0x84	0x0D	0x0A

Example: Stop Inventory

Command: A5 5A 00 08 8C 84 0D 0A

3.3.6 Stop Inventory Response

Length: 0x00 0x09
 Command: 0x8D
 Data: Result (0x01=OK, 0x00=failed)

Function: Stop Inventory response

Stop Inventory response frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x09	0x8D	0x01=OK 0x00=fail	0xXX	0x0D	0x0A

Example: Stop succeeded.

Command: A5 5A 00 09 8D 01 85 0D 0A

3.3.7 Read Tag

Length: 0x00 0xXX
 Command: 0x84
 Data: Access Password, Mask Bank, Mask Address, Mask Length, Data Bank, Address, Length
 Function: Read tag's memory

Read Tag command frame

SOF		Length		Cmd	Data			
0xA5	0x5A	0xXX	0xXX	0x84	AP (MSB)	AP	AP	AP (LSB)
Data								
MaskBank	MaskAddr (MSB)	MaskAddr (LSB)	MaskLen (MSB)	MaskLen (LSB)	MaskData (MSB)	...	MaskData (LSB)	
Data					CRC		EOF	
Bank	Addr (MSB)	Addr (LSB)	Len (MSB)	Len (LSB)	0xXX	0x0D	0x0A	

AP is the Access Password, Mask Bank (0x01=EPC, 0x02=TID, 0x03=USER), Mask Address in bits, Mask Length in bits (0x00=no mask), Mask Data bytes (right-padded with zeroes if Mask Length is not byte aligned), Data Bank, data start Address in words, data Length in words, Data words (multiple of 2 bytes).

Example 1: Access password is 0x55555555, no mask, read 3 words, TID bank, at address 2.

Command: A5 5A 00 16 84 55 55 55 55 00 00 00 00 02 00 02 00 03 91 0D 0A

Example2: Access password is 0x00000000, mask on TID bank, bit 2, 13 bits length, mask is 1110001000000'b, read 6 words from EPC bank, starting at address 2.

Command: A5 5A 00 18 84 00 00 00 00 02 00 02 00 0D E2 00 01 00 02 00 06 76 0D 0A

3.3.8 Read Tag Response

Length: 0x00 0xXX
 Command: 0x85
 Data: Result (0x01=OK, 0x00=failed), Error Code, Data Length, Data bytes.

Function: The data read from the tag's memory.

Read Tag response frame

SOF		Length		Cmd		Data	
0xA5	0x5A	0xXX	0xXX	0x85		Result	Error
Data					CRC	EOF	
Len (MSB)	Len (LSB)	Data (MSB)	...	Data (LSB)	0xXX	0x0D	0x0A

The amount of data read is specified by the data length parameter in Read Tag command. If the read failed, then the response frame shows no data; if the read succeeded, the Error is set to 0x00

Example: Success reading 3 words of TID bank data: 0x123456789abc

Command: A5 5A 00 12 85 01 00 00 03 12 34 56 78 9A BC BB 0D 0A

3.3.9 Write Tag

Length: Variable
 Command: 0x86
 Data: See the command frame description below.

Function: Write Data to the tag's memory

Write Tag command frame

SOF		Length		Cmd		Data				
0xA5	0x5A	0xXX	0xXX	0x86	APwd (MSB)	APwd	APwd	APwd	APwd (LSB)	
Data										
MaskBank		MaskAddr (MSB)	MaskAddr (LSB)	MaskLen (MSB)	MaskLen (LSB)	MaskData (MSB)		...	MData (LSB)	
Data								CRC	EOF	
Bank	Addr (MSB)	Addr (LSB)	Len (MSB)	Len (LSB)	Data (MSB)	...	Data (LSB)	0xXX	0x0D	0x0A

Access Password, Mask Bank (0x01 = EPC, 0x02=TID, 0x03=USER), Mask Start Address in bits, Mask Length in bits (0x00=no mask), Mask Data bytes (right-padded with zeroes if Mask Length is not byte aligned), Data Bank, data Start Address in words, data Length in words, Data words (multiple of 2 bytes, MSB first).

Example 1: No mask, write 6-words of data into EPC bank,
Data=0x00112233445566778899aabb, start address is 2, access password is
0x12345678.

Command: A5 5A 00 22 86 12 34 56 78 00 00 00 00 00 01 00 02 00
06 00 11 22 33 44 55 66 77 88 99 AA BB A9 0D 0A

Example 2: mask on TID, at address 0, mask length is 96 bits, write 6 words to
EPC bank, Data=0x00112233445566778899aabb, start address is 2, access password
is 0x00000000.

Command: A5 5A 00 2E 86 00 00 00 00 02 00 00 00 60 E2 00 34 14
01 33 01 00 10 38 D2 B5 01 00 02 00 06 00 11 22 33 44 55 66
77 88 99 AA BB 71 0D 0A

3.3.10 Write Tag Response

Length: 0x00 0x0A
Command: 0x87
Data: Result (0x01=OK, 0x00=failed), Error Code

Function: Write Tag response (the Error is 0x00 if the
command succeeded)

Write Tag response frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x87	0x01=OK 0x00=failed	Error	0xXX	0x0D	0x0A

Example: Write Data failed, the Error is 0x03

Command: A5 5A 00 0A 87 00 03 8E 0D 0A

3.3.11 Lock Tag

Length: Variable
 Command: 0x88
 Data: Access Password, Mask (Bank, Address, Length, Data), Lock data (3 bytes)

Function: Lock or Unlock a memory bank

Lock Tag command frame

SOF		Length		Cmd	Data							
0xA5	0x5A	0xXX	0xXX	0x88	APwd (MSB)	APwd	APwd	APwd (LSB)	Mask Bank			
Data										CRC	EOF	
Mask Addr (MSB)	Mask Addr (LSB)	Mask Len (MSB)	Mask Len (LSB)	Mask Data (MSB)	...	Mask Data (LSB)	Lock (MSB)	Lock	Lock	0xxx	0x0D	0x0A

Access Password (APwd), Mask Bank (0x01 = EPC, 0x02=TID, 0x03=USER), Mask Start Address in bits, Mask Length in bits (0x00=no mask), Mask Data bytes (right-padded with zeroes if Mask Length is not byte aligned).

Lock data is 3 bytes (24 bits), bits 0 to 9 are the Action bits, bits 10 to 19 are the Mask bits, bits 20 to 23 are not used.

Example: Mask on TID bank at address of 0 bits, mask length is 96 bits, mask data is 0xE2003414013301001038D2B5, lock EPC + RFU banks (0xFC2A0), password is 0x760039AD.

Command:

```
A5 5A 00 20 88 76 00 39 AD 02 00 00 00 60 E2 00 34 14 01 33
01 00 10 38 D2 B5 0F C2 A0 FB 0D 0A
```

3.3.12 Lock Tag Response

Length: 0x00 0x0A
 Command: 0x89
 Data: Result (0x01=OK, 0x00=failed), Error code

 Function: Lock or unlock tag memory bank response
 (Error is 0x00 if the command succeeded)

Lock Tag Response frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x89	0x01=OK 0x00=failed	Error	0xFF	0x0D	0x0A

Example: Lock succeeded.

Command: A5 5A 00 0A 89 01 00 82 0D 0A

3.3.13 Kill Tag

Length: 0x00 0xxx
 Command: 0x8A
 Data: KP (kill password), MMB, MSA, MDL, MData
 Function: Kill a tag.
 Kill Tag command frame

SOF		Length		Cmd	Data					
0xA5	0x5A	0xXX	0xXX	0x8A	KillPwd (MSB)	KillPwd	KillPwd	KillPwd (LSB)		
Data									CRC	EOF
Mask Bank	Mask Addr (MSB)	Mask Addr (LSB)	Mask Len (MSB)	Mask Len (LSB)	Mask Data (MSB)	...	Mask Data (LSB)	0xxx	0x0D	0x0A

Kill Password, Mask Bank (0x01 = EPC, 0x02=TID, 0x03=USER), Mask Start Address in bits, Mask Length in bits (0x00=no mask), Mask Data bytes (right-padded with zeroes if Mask Length is not byte aligned).

If the Kill Password written into the tag's memory is 0x00000000, then the tag will ignore the kill command, and the kill command will fail.

Example: Mask on the EPC bank at the address of 32 bits, mask length is 96 bits, mask data is 0x00112233445566778899AABB, kill password is 0x760039AD.

Command:

A5 5A 00 1D 8A 76 00 39 AD 01 00 20 00 60 00 11 22 33 44 55
 66 77 88 99 AA BB 34 0D 0A

3.3.14 Kill Tag Response

Length: 0x00 0x0A
 Command: 0x8B
 Data: Result (0x01=OK, 0x00=failed), Error Code
 Function: Kill tag response
 Kill Tag response frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x8B	0x01=OK 0x00=failed	Error	0xXX	0x0D	0x0A

Error is 0x00 if the command succeeded.

Example: Kill succeeded.

Command: A5 5A 00 0A 8B 01 00 80 0D 0A

3.3.15 Timed Inventory

Length: 0x00 0x0A
 Command: 0x90
 Data: Timeout, 10 - 30000ms (0x000A - 0x7530)
 Function: Repeat inventories until a timeout and report the number of tags found.

Send the 'Get Timed Inventory Result' command to retrieve the tag's EPC data.

Timed inventory frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x90	Timeout (MSB)	Timeout (LSB)	0xxx	0x0D	0x0A

Example: Perform inventories for 500 msec

Command: A5 5A 00 0A 90 01 F4 6F 0D 0A

3.3.16 Timed Inventory Response

Length: 0x00 0x0C
 Command: 0x91
 Data: Number of tags read (4 bytes)
 Function: Timed Inventory response (number of tags found)

Timed Inventory response frame

SOF		Length		Data				CRC	EOF	
0xA5	0x5A	0x00	0x0C	DByte3 (MSB)	DByte2	DByte1	DByte0 (LSB)	0xXX	0x0D	0x0A

Data (4 bytes) is the number of tags found, 0x00000000-0xFFFFFFFF (0-4294967295)

Example: 10 tags were found during a timed inventory

Command: A5 5A 00 0C 91 00 00 00 0A 97 0D 0A

3.3.17 Get Timed Inventory Result

Length: 0x00 0x08
 Command: 0x92
 Data: None
 Function: Get EPC data of tags inventoried during a timed inventory.

Get Timed Inventory result command frame

SOF		Length		Cmd	Data	CRC	EOF	
0xA5	0x5A	0x00	0x08	0x92	None	0xxx	0x0D	0x0A

Example: Get timed inventory result.

Command: A5 5A 00 08 92 9A 0D 0A

3.3.18 Block Write

Length: 0x00 0xxx
 Command: 0x93
 Data: See below

Function: Block Write the specific length of data at a specific address in the tag.

Block Write command frame

SOF		Length		Cmd		Data				
0xA5	0x5A	0xXX	0xXX	0x93	APwd (MSB)	APwd	APwd	APwd	APwd (LSB)	
Data										
MaskBank		MaskAddr (MSB)	MaskAddr (LSB)	MaskLen (MSB)	MaskLen (LSB)	MaskData (MSB)		...	MData (LSB)	
Data								CRC	EOF	
Bank	Addr (MSB)	Addr (LSB)	Len (MSB)	Len (LSB)	Data (MSB)	...	Data (LSB)	0xXX	0x0D	0x0A

Access Password, Mask Bank (0x01 = EPC, 0x02=TID, 0x03=USER), Mask Start Address in bits, Mask Length in bits (0x00=no mask), Mask Data bytes (right-padded with zeroes if Mask Length is not byte aligned), Data Bank, data Start Address in words, data Length in words, Data words (multiple of 2 bytes, MSB first).

Example: Tag PC=0x3000, EPC=0xE2003411B802011383258566, write 6-words to EPC bank at address 2, Data=0x00112233445566778899aabb, password is 0x74290fd8

Command:

```
A5 5A 00 2B 93 74 29 0f d8 30 00 E2 00 34 11 B8 02 01 13 83
25 85 66 01 00 02 00 06 00 11 22 33 44 55 66 77 88 99 AA BB
2D 0D 0A
```

3.3.19 Block Write Response

Length: 0x00 0x0A
 Command: 0x94
 Data: Result (0x01=OK, 0x00=failed), Error Code

Function: Block Write response

Block Write response frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x94	0x01=OK 0x00=failed	Error	0xXX	0x0D	0x0A

Error is 0x00 if command succeeded

Example: Write failed, Error is 0x01

Command: A5 5A 00 0A 94 00 01 9F 0D 0A

3.3.20 Block Erase Tag

Length: 0x00 0xxx
 Command: 0x95
 Data: See below
 Function: Block Erase tag's memory

Block Erase command frame

SOF		Length		Cmd		Data		
0xA5	0x5A	0xXX	0xXX	0x95	APwd (MSB)	APwd	APwd	APwd (LSB)
Data								
MaskBank	MaskAddr (MSB)	MaskAddr (LSB)	MaskLen (MSB)	MaskLen (LSB)	MaskData (MSB)	...	MData (LSB)	
Data						CRC	EOF	
Bank	Addr (MSB)	Addr (LSB)	Len (MSB)	Len (LSB)		0xXX	0x0D	0x0A

Access Password, Mask Bank (0x01 = EPC, 0x02=TID, 0x03=USER), Mask Start Address in bits, Mask Length in bits (0x00=no mask), Mask Data bytes (right-padded with zeroes if Mask Length is not byte aligned), Erase Bank, Erase Start Address in words, Erase Length in words (multiple of 2 bytes, MSB first).

Example: Tag PC=0x3000, EPC=0xE2003411B802011383258566, erase 6 words of EPC bank at the address 2, password is 0x74290fd8

Command: A5 5A 00 1F 95 74 29 0f d8 30 00 E2 00 34 11 B8 02 01 13 83 25 85 66 01 00 02 00 06 1F 0D 0A

3.3.21 Block Erase Tag Response

Length: 0x00 0x0A
 Command: 0x96
 Data: Result (0x01=OK, 0x00=failed), Error Code
 Function: Block Erase tag response.

Block Erase tag response frame

SOF		Length		Cmd		Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0x96		0x01=OK 0x00=failed	Error	0xXX	0x0D	0x0A

Error is 0x00 if command succeeded

Example: Block erase succeeded, error is 0x00.

Command: A5 5A 00 0A 96 01 00 9D 0D 0A

3.3.22 Operation Failed Response Frame

Length: 0x00 0x0A
 Command: 0xFF
 Data: Error code (2 bytes)

Function: Operation failed response

Operation failed response frame

SOF		Length		Cmd	Data		CRC	EOF	
0xA5	0x5A	0x00	0x0A	0xFF	Error code (MSB)	Error Code (LSB)	0xFF	0x0D	0x0A

Error codes:

0001 Inventory failed
 0002 CRC error
 0003 Temperature is too high
 0004 Reflected power is too high

Example: Error Code is 0x0001

Command: A5 5A 00 0A FF 00 01 F4 0D 0A