AN12305

APDU Specification of A71CL Security Module

Rev. 1.0 — 14 December 2018 Application note
515410 COMPANY PUBLIC

Document information

Information	Content
Keywords	Security Module, A71, APDU
Abstract	This document provides the APDU API description for the A71CL security module.

1 Introduction

1.1 Scope

This specification describes the CL applet APDU interface to IoT devices, based on the A710x family[2] with JCOP 2.4.2 R1. It mainly supports device ID storage, sensitive data storage, cryptography, signature and message digest.

A71CL also supports provisioning and personalization services without involving card-manager privileges using applet-specific keys.

1.2 Architecture

As shown in the architecture diagram below, A71CL consists of a crypto module and a security storage module.

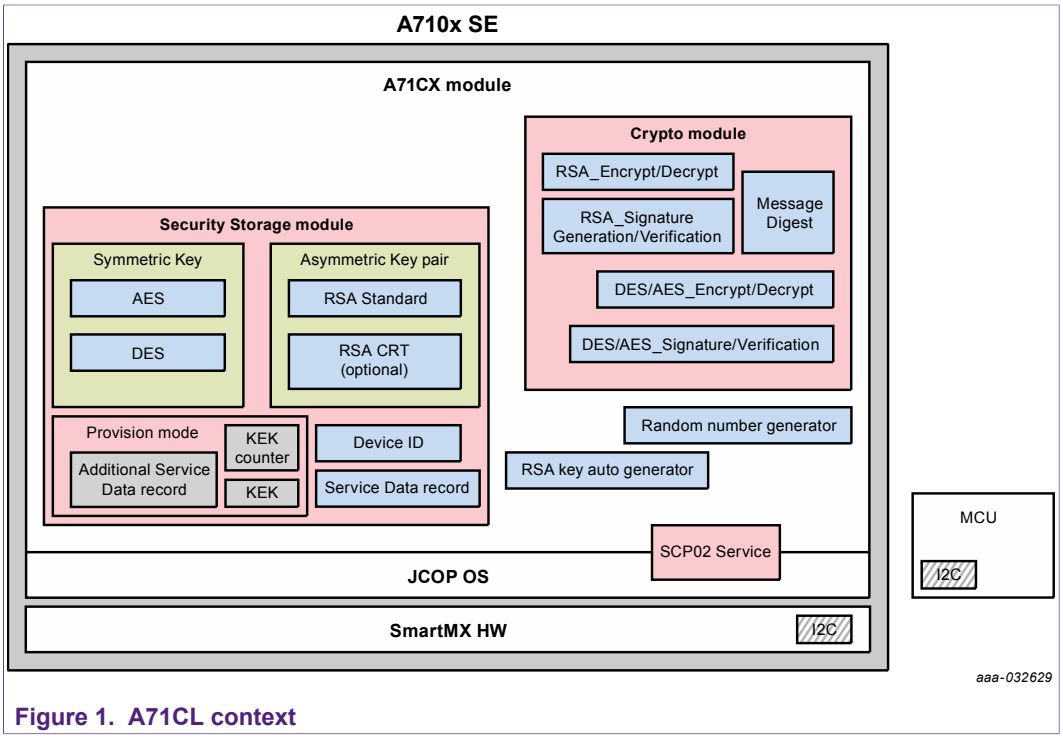

Figure 1. A71CL context

The security storage module acts as a data container. Sensitive data like Device ID, Service Data, and keys are stored under SCP or RFC 5649 protection.

The crypto module provides cryptographic operations using Cryptography Commands (APDUs).

The provision mode is an alternative way to personalize sensitive data when the card manager is disabled. This service is activated through installation parameter.

1.3 Product Basic Features

The A71CL Security Module provides the following functionality based on A71 platform:

2 The Hardware product details can be found in the document sds_495123_A71CL.

- Message Digest with SHA1, SHA224, SHA256.
- Random number generator.
- Symmetric Key Storage: one DES key or AES key.
- Asymmetric Key Storage: one RSA Standard keypair or one RSA CRT keypair.
- Auto RSA key generator ranges from 512-bit key length to 2048-bit key length. Including either RSA or RSA CRT.
- Symmetric encryption/decryption with DES_CBC_NOPADDING, DES_ECB_NOPADDING, AES_CBC_NOPADDING, AES_ECB_NOPADDING.
- Symmetric signature generation/verification with DES_CBC_ISO9797_M1, DES_CBC_ISO9797_M2, AES_CBC_ISO9797_M1, AES_CBC_ISO9797_M2.
- Asymmetric encryption/decryption with RSA_NOPADDING, RSA_ PKCS1.
- Asymmetric signature generation /verification with RSA_SHA1(PKCS1), RSA_SHA256.
- Service data storage: Service data record read and write by SCP protection.
- Device ID value in secure storage.
- SCP 02 service with option “i” = ‘55’[3].

1.4 Product Extended Features

Compared to the default mode when the Card Manager is enabled, the A71CL Security Module provides an extended feature called provision mode. This mode provides functions even the Card Manager is disabled

- An additional service data record which can be free read and write is supported[4]. It is not recommended to store any confidential data.
- Service data record, the additional service data record and key can either be stored in plain text directly or protected by SCP protection or by RFC5649.
- Service data record, the additional service data record and key can be deleted securely[5].

1.5 A71CL Memory View

Table 1. Memory view

A71CL variant	Applet in EEPROM of A71
Number of symmetric keys	1
Number of Asymmetric keys	1
Service data storage size	up to 5 KB[1]

[1] This value is calculated under A71 with only A71CL applet on EEPROM without security domain.

3 Refer to E Secure Channel Protocol '02' in GlobalPlatform Card Specification 2.2.1
4 This is supported in Plain Injection Mode
5 This is supported in Full Reset Mode

1.6 Platform Characteristics

1.6.1 APDU Interface

The A71CL deploys an APDU interface as defined in [ISO/IEC 7816-4:2005].

The A71CL supports only standard APDUs. Extended APDUs are not supported.

1.6.1.1 Maximum APDU size

APDUs have a command data payload of maximum 255 bytes and a response data payload of maximum 255 bytes. This limitation applies to all commands when SCP is not active (see Section 2.5).

When SCP is active the maximum command payload is 239 bytes and the maximum response data payload is 239 bytes.

These limitations are applicable both for the Command APDU and the Response APDU.

2 A71CL Configuration

2.1 Provisioning Mode Configuration

Considering the Card Manager may be disabled in some cases. The A71CL offers four provision modes which can be configured by NXP.

1. **Default Mode:**
   In this mode, the card manager is enabled. SCP keys are required to manage sensitive data.
2. **Plain Injection Mode:**
   In this mode the ID and keys can get personalized in plain communication without additional protection data against disclosure or manipulation. Plain Injection mode should only be used in trusted provision.
   Before A71CL is delivered to the end user, this mode shall be disabled by the Disable Plain Injection Mode command.
3. **Authentication Mode:**
   In this mode the ID and keys can get personalized encrypted using the KEK (Key Encryption Key) when the SCP keys are not available. Sensitive data wrapped by KEK using RFC5649 can only be added or replaced when it has been successfully unwrapped on the card.
4. **Full Reset Mode:**
   This mode is the same as “Authentication Mode” plus the option to reset user data. It allows user data such as service data, the additional service data and key objects to be reset or wiped. This is useful during development or to assure that the applet is in an initial state before provisioning.

Table 2. Mode configure parameters set in ‘C9’ tag of INSTALL COMMAND parameter:

Tag	Length	Value	Description
0x80 Mode Indicator	0x01	0x00	**Default Mode**
	0x01	0x01	**Plain Injection Mode**
	0x10	Default KEK with 16 bytes AES key	**Authentication Mode**
	0x11	Default KEK with 16 bytes AES key +(1 byte) 0x5A	**Full reset Mode**
0xAC Restriction Indicator	0x01	0x80	**1xxx xxxxb RSA CRT key type is disabled.** 0 xxx xxxxb RSA CRT key type is abled.

2.1.1 Mode Configuration Flow

2.1.1.1 Plain Injection Mode

When A71CL is in Plain Injection Mode, sensitive data can be written freely, until it is disabled (see Disabled “Plain Injection Mode”).

Note:

1. *This is a convenience way for provisioning. Because the sensitive data can be written freely without any authentication, on security concern, this mode shall be disabled once provisioned before handing over to user.*
2. *Except for GenerateKeyPair, SecurityStorage, GetChallenge, GetID and GetVendorInfo, all other functional commands are disabled in this mode, until plain injection is disabled.*
3. *When this mode is enabled, the commands Disable Plain Injection Mode, Put Data and Free Read Service Data are activated.*
4. *Using the Disable Plain Injection Mode command changes the mode to "Default Mode" which disables this command together with Put Data. At the same time other functional commands are activated.*

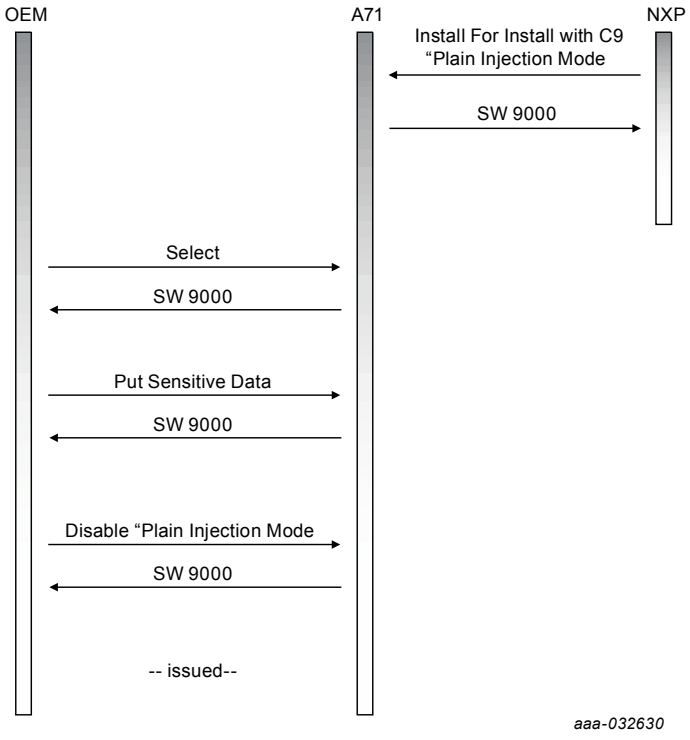

Figure 2. Example Plain Injection Mode

2.1.1.2 Authentication Mode

Authentication mode is more secure than plain injection mode. In this mode, key objects, service data or the additional service data wrapped by KEK using RFC5649 can only be added or replaced after being successfully unwrapped on the A71CL application.

Note:

1. *The default KEK used for unwrapping is initially set in the C9 installation parameter.*
2. *The new KEK key to be updated by the OEM shall be wrapped by the original old KEK using Put Data.*
3. *Authentication Mode is incompatible with Plain Injection Mode. When Authentication Mode is set through C9 installation parameter, the sensitive data could only be stored after the RFC5649 unwrap is succeeded.*

4. *All the key data or service data to be added or updated in this mode shall be wrapped by RFC5649 using KEK before sending.*
5. *The key data and the service data record are written by Security Storage (included in data field). The additional service data record is written by Put Data. The command A71CL_External_Authentication must be executed before using the Security Storage read function.*
6. *GenerateKeyPair's data field shall also be wrapped by RFC5649 using KEK before sending.*
7. *When this mode is enabled, the command A71CL_External_Authentication, Put Data, Free Read Service Data and other commands are activated.*

The following flow is the example showing how the sensitive data stored into A71CL.

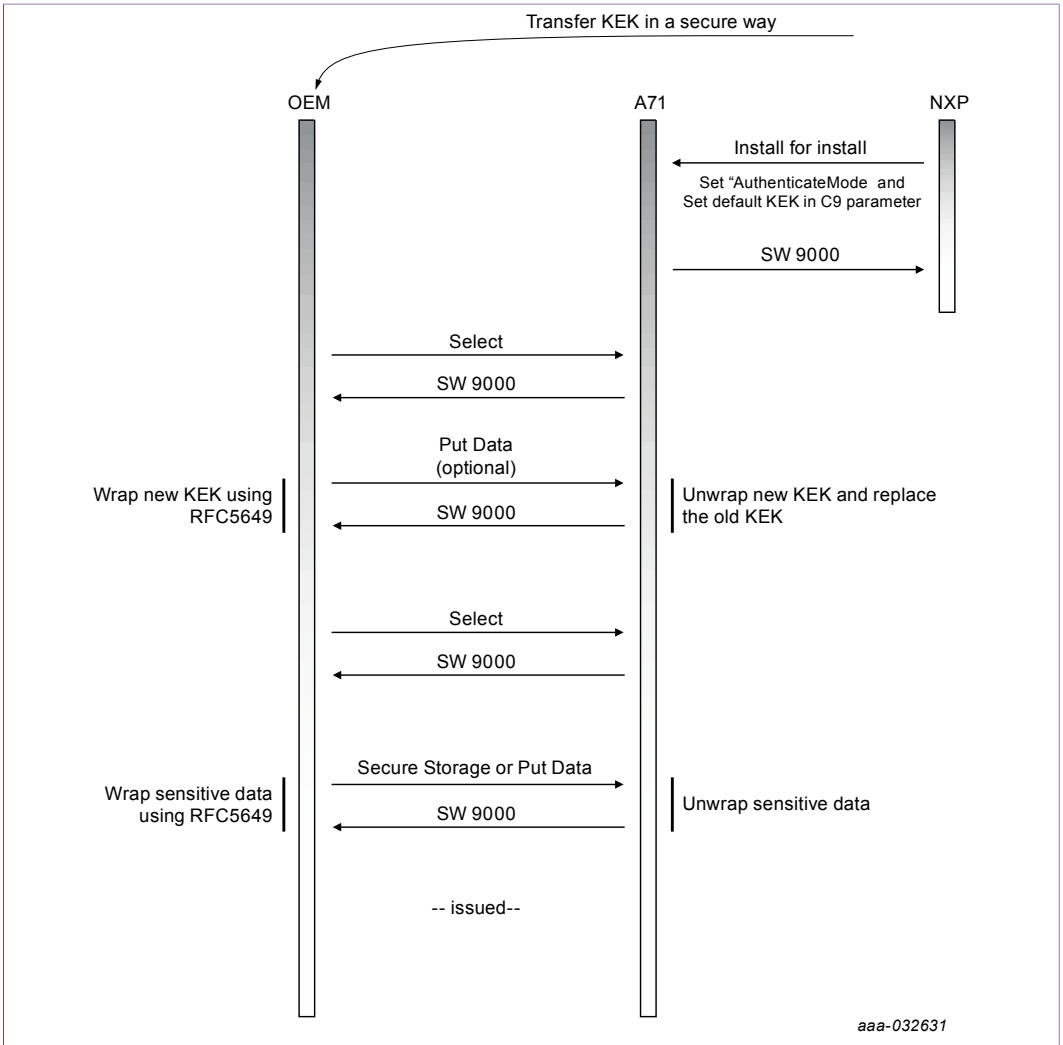

Figure 3. Example Authentication Mode

2.1.1.3 Full Reset Mode

When it is not feasible to re-instantiate the applet, Full Reset mode allows them reset / delete user data such as service data and key object.

Note:

1. *Full reset mode is an extension of the authentication mode and has all features of authentication mode.*
2. *This mode can be set through the C9 installation parameter.*
3. *This mode shall be set under authentication mode. When Full Reset mode is set the Full Reset, command is activated.*
4. *All user data, including key objects, are permanently deleted when the Full Rest command is executed.*

The flow below shows how to perform a full-reset:

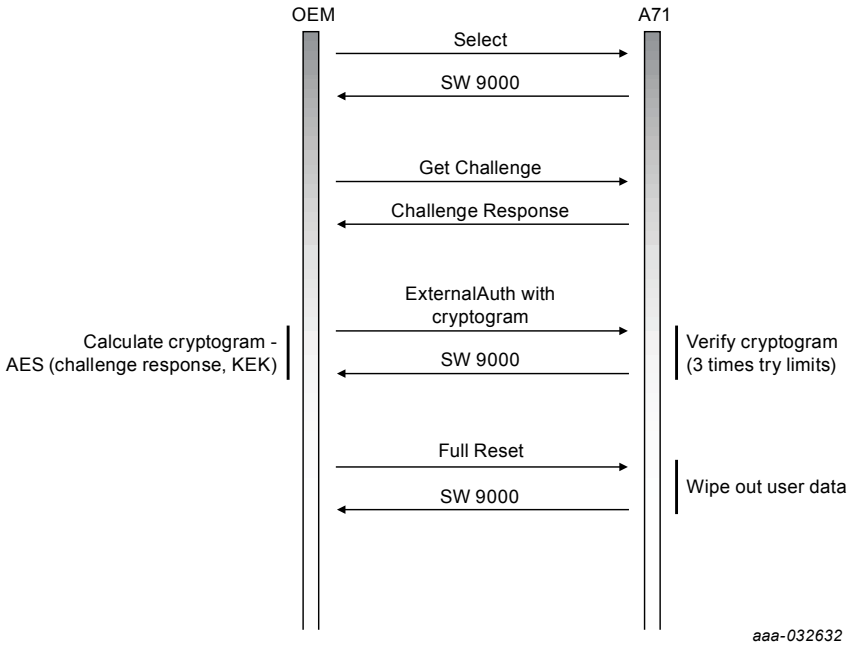

Figure 4. Example Full Reset Mode

2.1.2 APDU Command Reference

The following commands are all APDU commands supported under each corresponding mode.

Table 3. A71CL APDU commands

Command	Default Mode[1]	Plain Injection Mode	*Authentication Mode*	Full reset Mode
Disable Plain Injection Mode		x		
Put Data		x	x	x
Free Read Service Data	x	x	x	x
A71CL_External_Authentication			x	x
Full Reset				x
GetChallenge	x	x	x	x
SecurityStorageData	x	x	x	x
ComputeDigest	x		x	x

Command	Default Mode[1]	Plain Injection Mode	*Authentication Mode*	Full reset Mode
GenerateKeyPair	x	x	x	x
SymmetricEncrypt	x		x	x
AsymmetricEncrypt	x		x	x
GetID	x	x	x	x
GetVendorInfo	x	x	x	x

[1] Default mode also included the mode that previously switched from Plain Injection Mode.

2.2 Initial State

The initial state for all the object storing sensitive data is null unless personalized.

The initial state for mode configuration shall be set during the installation depending on the customers, else the installation will be failed.

The initial state for KEK in authentication mode and full reset mode includes:

1. The KEK key value is set in the installation parameter;
2. The try limit counter is set to 3.

2.3 Life Cycle

This section defines the following states applicable to A71CL:

1. UNPERSONALIZED
2. PERSONALIZED

The state is changed implicitly.

The A71CL life cycle coding state can be obtained through the associated Security Domain by the Global Platform command GET STATUS.

2.3.1 UNPERSONALIZED State:

This state indicates that there's no symmetric or asymmetric key personalized and cryptography commands are not allowed. After the symmetric or asymmetric key is personalized by Sensitive Data Storage commands, the state switches to PERSONALZIED automatically.

A71CL application returns to UNPERSONALZIED automatically if an error occurs during the key personalization, or if triggered by the Full Reset command in the PERSONALIZED state.

2.3.2 PERSONALIZED State:

When either a Symmetric or Asymmetric key is personalized the A71CL life cycle state switches from UNPERSONALIZED to PERSONALIZED.

This state indicated that the cryptography commands are allowed.

2.3.3 Life Cycle Coding [6]

The Life cycle has a bit-oriented code value on one byte as described in the following table. The code value can be read from the associated Security Domain by the Global Platform command GET STATUS.

Table 4. Life cycle coding

B8	B7	B6	B5	B4	B3	B2	B1	Meaning
0	0	0	0	0	0	1	1	UNPERSONALIZED
0	0	0	1	1	1	1	1	PERSONALIZED

2.4 Example Mode Configuration

Below are examples of how to set each mode. The applet is installed by NXP, the parameters here are only for reference for particular product configurations:

- **Default**: '800100'.
- **Plain Injection mode**: '800101'.
- **Authentication mode**: '8010000102030405060708090A0B0C0D0E0F'.
  - '80' refers to Tag for Mode Indicator.
  - '10' refers to length.
  - '000102030405060708090A0B0C0D0E0F' refers to the default KEK value.
- **Full reset mode**: 8011000102030405060708090A0B0C0D0E0F5A
  - '80' refers to Tag for Mode Indicator.
  - '10' refers to length.
  - '000102030405060708090A0B0C0D0E0F' refers to the default KEK value.
  - '5A' refers to full reset mode support.

2.5 Secure Channel Protocol (SCP)

The Security Module is connected to the Host CPU using, for example, an I2C link employing the SCI2C protocol (compare to [SCI^{2}C] and [A710x]).

The Host to Security Module communication is optionally protected by a Secure Channel Protocol (SCP) according to the [Global Platform SCP02] specification, using "i" = '55':

- Initiation mode explicit,
- C-MAC on modified APDU,
- ICV set to zero,
- ICV encryption for C-MAC session,
- 3 Secure Channel Keys of 128 bits,
- well-known pseudo-random algorithm (card challenge),
- no R-MAC.

2.5.1 Secure Channel Initiation

The Secure Channel initiation is done under the A71CL application by the SCP command pair INITIALIZE UPDATE and EXTERNAL AUTHENTICATE.

6 Life Cycle Coding is not acceptable for A71CL.

2.5.2 Secure Channel APDU Commands

Table 5 summarizes the minimum-security requirements for the APDU commands.

Table 5. Minimum-security

Command	Minimum-security
SecurityStorage	C-MAC
Generate Key Pair	Secure Channel initiation
Oher commands	None

2.5.3 Secure Channel Error Condition

Table 6 summarizes the error code Secure Channel initiation or unwrapping may return.

Table 6. Error Condition

SW1	SW2	Comment
0x63	0x00	Authentication of host cryptogram failed
0x67	0x00	Wrong length
0x69	0x82	Security is not satisfied
0x6A	0x88	Referenced data not found
0x90	0x00	Executed correctly

2.6 Security Feature

2.6.1 ID Integrity Protection

The Device ID is a character string which links to the unique identifier of equipment, corresponding key, certificate and ID² Server. It is solidified in an element chip and is resistant to tampering and prediction, and globally unique.

2.6.1.1 The Structure of ID Value

The Device ID value in A71CL is rommified during the first writing in the personalization. The length of ID depends on the user.

The structure below shows how ID is stored.

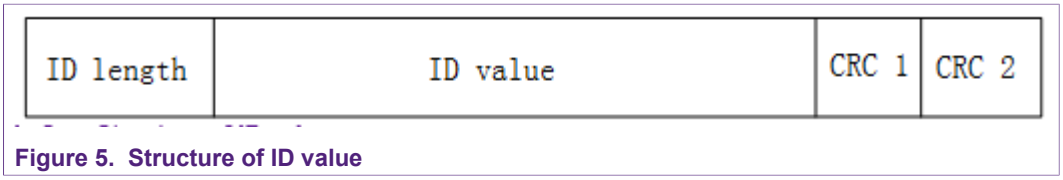

Figure 5. Structure of ID value

2.6.1.2 Integrity Protection

During the personalization, an Device ID value will be injected into the A71CL applet as well as its checksum.

The A71CL applet checks the CRC each time the ID value is read or used to see if it has been attacked.

2.6.2 KEK Protection

The authentication and full reset modes exists have a counter for KEK protection.

If the A71CL_External_Authentication fails 3 times (if the counter decreases to 0), all functionality using the KEK key is rejected with SW_DATA_INVALID.

2.6.3 Personalization Restriction

The following personalization restriction apply to A71CL applet:

1. GenerateKeyPair cannot be executed as long as the SecurityStorageData command has been executed successfully or the asymmetric key already exists;
2. When the state is in UNPERSONALIZED , the commands SymmtricEncrypt, ComputeDigest and AsymmtricEncrypt cannot be executed.

3 A71CL Coding Rules

3.1 General Coding Rules

3.1.1 A71CL Security Application Instruction Set

A71CL security application instruction set is as listed in Table 7:

Table 7. Instruction sets

No	Instruction Name	CLA	INS	Function Description	Compatibility
1	GetChallenge	00	84	Take random number	A71CL Proprietary
2	SecurityStoreData	84	E2	R/W instruction in secure channel	A71CL Proprietary
3	ComputeDigest	80	F0	Calculate digest value (SHA1 / SHA256 / SHA 512/ SM3)	A71CL Proprietary
4	GenerateKeyPair	80	F2	Generate key pair	A71CL Proprietary
5	AsymmetricEncrypt	80	F4	Asym. algorithms	A71CL Proprietary
6	SymmetricEncrypt	80	F6	Sym. Algorithm (3DES / AES)	A71CL Proprietary
7	GetID	80	F8	Get ID	A71CL Proprietary
8	GetVendorInfo	80	FC	Get vendor information	A71CL Proprietary
9	Initialize Update	80	50	To initiates the initiation of a Secure Channel Session	GP Proprietary
10	External Authenticate	84	82	To authenticate the host	GP Proprietary
11	Disable Plain Injection Mode	80	FE	Close the plain injection mode	A71CL Proprietary
12	Put Data	84	12	Put service data or replace KEK key	A71CL Proprietary
13	Free Read Service Data	80	71	To be freely read service data	A71CL Proprietary
14	External Authenticate	00	82	Used to authenticate by A71CL applet	A71CL Proprietary
15	Full Reset	80	EA	Clear all user data	A71CL Proprietary

3.1.2 Key Type

Key type descriptions are as listed in Table 8:

Table 8. Key type

Key type	Value	Description
3DES	‘00‘	Triple-DES key, symmetric algorithm
AES	‘01‘	AES key, symmetric algorithm
RSA_Standard	‘02’	RSA standard key, asymmetric algorithm
RSA_CRT	‘03’	RSA Chinese remainder theorem key, asymmetric algorithm

3.1.3 Key Identifiers

Key element identifiers are as listed in Table 9:

Table 9. Definitions of Security Application Key Storage Identifiers

Type	Flag	Length (byte)	Value
3DES	'40'	'10'or '18'	Key value
AES	'41'	'10'or '18'or'20'	Key value
RSA-CRT-INVQ	'49'	Key mod length/2	INVQ value
RSA-CRT-DP	'50'	Key mod length/2	DP value
RSA-CRT-DQ	'51'	Key mod length/2	DQ value
RSA-D	'64'	Private key value length	Private key value
RSA-E	'65'	'04'	Public key exponent
RSA-N	'6E'	Public key value length	Public key mod value
RSA-CRT-P	'70'	Key mod length/2	Prime P value
RSA-CRT-Q	'71'	Key mod length/2	Prime Q value

3.1.4 A71CL Configuration Options

A71CL configuration parameter called 'i' defines the functions the A71CL module supports in the form of 4-byte bitmaps.

B3-B2 and B0 is defined as follows: (byte 0)

Table 10. A71CL Configuration Option B0 of 'i'

b8	b7	b6	b5	b4	b3	b2	b1	Description
-	-	-	-	-	-		1	3DES algorithm supported
-	-	-	-	-	-	1		AES algorithm supported

B1 is defined as follows:

(byte1)

Table 11. A71CL Configuration Option B1 of 'i'

b8	b7	b6	b5	b4	b3	b2	b1	Description
-	-	-	-	-	-		1	RSA algorithm supported
-	-	-	-	-	-	1		RSA CRT algorithm supported

B2 is defined as follows:

(byte 2)

Table 12. A71CL Configuration Option B2

b8	b7	b6	b5	b4	b3	b2	b1	Description
-	-	-	-	-			1	SHA-1 algorithm supported
-	-	-	-	-		1		SHA-224 algorithm supported
-	-	-	-	-	1			SHA-256 algorithm supported

3.1.5 Status Word

SW1 SW2 is the return code of the application execution command. The return information of any command is composed of at least one status word. The return data field is optional.

The status words are described in Table 13.

Table 13. Status Words

SW1	SW2	Description
90	**00**	Executed correctly
61	**xx**	Expected return data length of ISO7816 T0 protocol
62	**81**	The returned data may be wrong.
62	**83**	Selected file invalid, file or key validation error
63	**Cx**	x means the number of re-try times
63	**10**	There is still data not returned
64	**00**	Status flag not changed
65	**81**	Write EEPROM unsuccessfully
67	**00**	Wrong length
69	**00**	CLA does not match line protection requirements.
69	**01**	Invalid status
69	**81**	The command is incompatible with the file structure.
69	**82**	Does not meet the secure state
69	**83**	The key is locked.
69	**84**	No random number
69	**85**	Conditions of use are not satisfied.
69	**86**	The selected file is not available.
69	**87**	No security message
69	**88**	Data item in security message is incorrect.
6A	**80**	Data structure error/signature verification failure
6A	**81**	Function not supported
6A	**82**	File not found
6A	**83**	Record not found
6A	**84**	Lack of space
6A	**86**	Parameter P1 P2 error
6B	**00**	The file ends before the Le / Lc byte is reached; the offset is incorrect.
6C	**xx**	Le error
6D	**00**	Instruction code not supported
6E	**00**	Invalid CLA
6E	**01**	Invalid command sequence
6E	**02**	No secure environment or invalid secure environment

SW1	SW2	Description
6F	**00**	Invalid data
93	**03**	Application locked.
94	**01**	Algorithm not supported
94	**02**	Key type not supported
94	**03**	Key not found
94	**04**	ID input
94	**05**	The key type has existed
94	**06**	Required MAC is not available.
95	**xx**	XX indicates the number of bytes to be transmitted

4 A71CL APDU Interface

4.1 APDU Overview

There are six classified commands for the A71CL:

- Global Platform commands
- Sensitive Data Storage commands
- Cryptography commands
- Debug commands
- Read Information commands

This chapter explains these five classes in more detail.

4.1.1 Global Platform Commands

A71CL owns two Global Platform commands which are used to initiate secure channel.

Table 14. Global Platform commandsRe

Function	Description
GP_INITIALIZE_UPDATE	See GP_InitializeUpdate
GP_EXTERNAL_AUTHENTICATE	See GP_ExternalAuthenticate

4.1.2 Sensitive Data Storage Commands

A71CL has specific commands that can store sensitive data (key or service data).

Table 15. Sensitive Data Storage commands

Function	Description
SecurityStorageData	See *SecurityStorageData*
GenerateKeyPair	*See GenerateKeyPair*
Put Data[1]	See ***Put** Data*

[1] The Commands Put Data and Free Read Service Data act on “additional service data” but cannot act on “service data” at another memory location, whereas this is the other way round for ID2_SecurityStorageData”

4.1.3 Cryptography Commands

A71CL offers commands for cryptographic operations.

Table 16. Cryptography commands

Function	Description
ComputeDigest	See ComputeDigest
SymmetricEncrypt	See SymmetricEncrypt
AsymmetricEncrypt	See AsymmetricEncrypt

4.1.4 Debug Commands

A71CL offers commands that can reset all key object, service data, and other user data.

Table 17. Debug commands

Function	Description
Full Reset	See Full Reset
A71CL_External_Authentication	See A71CL_External_Authentication
Disable Plain Injection Mode	See Disable Plain Injection Mode

4.1.5 Read Information Commands

A71CL offers commands that can retrieve information on SE.

Table 18. Read Information commands

Function	Description
GetChallenge	See GetChallenge
GetID	See GetID
Free Read Service Data	See Free Read Service Data
GetVendorInfo	See GetVendorInfo

4.2 APDU Instruction Coding

4.2.1 A71CL Application Instruction

4.2.1.1 Get Challenge (Retrieve Random Number) Command

4.2.1.1.1 Definition and scope

The Get Challenge command is used to request a random number for the A71CL application's security process.

The random number can only be used for the next instruction. Whether the next command uses the random number or not, the random number will become invalid immediately.

4.2.1.1.2 Pre-condition

Secure message wrapped by SCP [7] can be used in default mode.

4.2.1.1.3 Post-condition

-

4.2.1.1.4 Command message

The command messages are as shown.

Table 19. Get Challenge Command message

Code	Value
CLA	0x00

7 Secure message refers to the data message wrapped by SCP.

Code	Value
INS	0x84
P1	0x00
P2	0x00
Lc	Not existing
Data	Not existing
Le	'04'~'10'

4.2.1.1.5 Command message data field

Command message data field does not exist.

4.2.1.1.6 Response message data field

Thee response message data field is the random number expected to return.

4.2.1.1.7 Response message status code

Table 20. Get Challenge– Status

SW1	SW2	Comment
0x67	0x00	Wrong length
0x6A	0x86	Parameter P1 P2 error
0x90	0x00	Executed correctly

See Secure Channel Error Condition for additional warning status code from SCP that A71CL may send back.

4.2.1.2 Compute Digest Command

4.2.1.2.1 Definition and scope

The Compute Digest command calculates a message digest of the data provided to the command using the specified digest algorithm provided by the command parameters. The 'to-be-calculated' data is sent to A71CL application by one or more Compute Digest commands. Upon receipt of all 'to-be-calculated' data blocks. The SE returns a fixed length digest of the data.

4.2.1.2.2 Pre-condition

Secure message can be used in default mode.

A71CL must be in state PERSONALIZED".

4.2.1.2.3 Post-condition

-

4.2.1.2.4 Command message

The Compute Digest command messages are as shown.

Table 21. Compute Digest Command message

Code	Value
CLA	0x80
INS	0xF0
P1	Block number (from 0)
P2	01: the last data; 00: the cascaded data (not the last one).
Lc	Length of data to be processed
Data	Data to be processed
Le	Not existing or the length of message digest value expected to return

4.2.1.2.5 Command message data field

The command message data field contains the data to be calculated.

Format definition:

When P1=0:

The command message data field is as shown in Table 22

Table 22. Compute Digest Command Message Data

Definition	Number of bytes	Description
Type	1	Digest algorithm type: 00: SHA1 01: SHA224 02: SHA256
Data	Lc-1	Block data defined in P1 to be digested

When P1 != ‘00’:

The command message data field is as shown in below table:

Table 23. Command Message Data Field

Definition	Number of bytes	Description
Data	Lc	Data to be calculated

4.2.1.2.6 Response message data field

If the current command is not the last message digest calculation command, the response data field is empty;

If the current command is the last message digest calculation command, the response message is the calculated message digest value.

4.2.1.2.7 Response message status code

Table 24. Compute Digest– Status

SW1	SW2	Comment
0x6A	0x80	Wrong data
0x6A	0x86	Parameter P1 P2 error

SW1	SW2	Comment
0x6E	0x01	Chaining is invalid
0x94	0x01	Algorithm is not supported
0x90	0x00	Executed correctly

Where A71CL does not support digest algorithm specified by data field, the command returns the status word '9401'.

See Secure Channel Error Condition, the additional warning status code from SCP that A71CL may send back.

4.2.1.3 SecurityStorage (Security Data Operation) Command

4.2.1.3.1 Definition and scope

The SecurityStorage command is used to write the device sensitive data stored in the SE and read no-sensitive-related data.

Sensitive data include th Device ID, keys and other associated sensitive data.

4.2.1.3.2 Pre-condition

Secure message shall be used in default mode.

In Default Mode, this command must establish secure channel before it executed.

In Authentication Mode or Debug Mode this command's data field needs to be wrapped by KEK key using RFC5649 first.

For a read instruction A71CL_External_Authentication must have been executed successfully.

The KEK counter must be below its max try limit.

4.2.1.3.3 Post-condition

1. The Device ID value once written cannot be updated.
2. The key data can be updated as long as the ID in the data field is equal to the existing ID.
3. The lifecycle transit to PERSONALIZED.

4.2.1.3.4 Command message

The SecurityStorage messages are as shown

Table 25. SecurityStorage Command message

Code	Value
CLA	0x84
INS	0xE2
P1	See P1 parameter description
P2	See P2 parameter description
Lc	Data field length
Data	Personalization data

Code	Value
Le	Not existing

P1, the read/write and block number control parameter, is described as follows:

Table 26. Definition of most significant bit “Bit 7” of P1

Bit7	Description
1	Read instruction
0	Write instruction

Table 27. Definition of secondary significant bit "Bit 6" of P1:

Bit6	Description
1	Service data instruction
0	Key operation instruction

Table 28. Definition of P1 Bit5~Bit0:

Bit5~Bit0	Description
‘XX’	Value range of the block number of cascaded data: "00"~"20"

Table 29. P2 parameter is cascade identification, which is defined as follows:

Bit5~Bit0	Description
‘01’	The last data to be processed
‘00’	Cascade data to be processed

4.2.1.3.5 Command message data field

P1-Bit7 = 0, indicating write instruction:

The data format to be written into the command message data field is defined as follows:

a) If the data field is a key:

Key header data format:

Table 30. SecurityStorage Key header data format

Definition	Byte number	Description
Device ID length	‘1’	
Device ID	X	Device ID data
KEY_TYPE	1	See Table 8
KEY_ID	1	‘00’: indicating A71CL.Key ‘01’~‘FF’: indicating business-related key
KEY	See table below	

KEY data field format is defined as follows:

Table 31. SecurityStorage Key data field format

Definition	Byte number	Description
KeyEl01 Tag	1	Tag of the first key element, as seen in Table "Definitions of Security Application Key Storage Identifiers "
KeyEl01 Length	2	Length of the first key element
KeyEl01 Value	KeyEl Length	Value of the first key element
KeyEl02 Tag	1	Tag of the second key element, as is seen in Table "Definitions of Security Application Key Storage Identifiers "
KeyEl02 Length	2	Length of the second key element
KeyEl02 Value	KeyEl Length	Value of the second key element
…		
KeyEln Tag	1	Tag of the nth key element, as is seen in Table 9 "
KeyEln Length	2	Length of the nth key element
KeyEln Value	KeyEl Length	Value of the nth key element

Note:

1. If key data are loaded by chaining, the first data include Key header data and the first package of key value content and the following ones include only the key value;

2.When the length of the Key header data + key data value content is less than 256-byte length, all the key data must be loaded at one time.

3. KeyEl 01~KeyEln must be components of the same key.

b) When it comes to the non-key service data:

Table 32. Non-key Service Data

Definition	Byte number	Description
Data	Lc	Service data

Note:

1. Read instruction data field does not exist.

2. Key-related business correlation settings and data are placed in the service data and the data details are determined by communication between the 3rd party and the OEM. (For example, key attempt limit, key management and maintenance authority, except A71CL key.)

4.2.1.3.6 Response message data field

The response message data do not exist under the write instruction.

The response message data are business associated data under the read instruction.

Note:

Read key data by read instruction is not possible.

4.2.1.3.7 Response message status code

Table 33. SecurityStorage – Status

SW1	SW2	Comment
0x69	0x82	Security is not satisfied
0x69	0x84	Data is invalid
0x6A	0x81	Functionality is not supported
0x6A	0x84	Out of memory
0x6E	0x01	Chaining is invalid
0x90	0x00	Executed correctly
0x94	0x01	Algorithm is not supported

See Secure Channel Error Condition for information on the additional SCP warning status code A71CL may return.

4.2.1.3.8 Restriction for service data:

1. The service data should be written or read in chain, otherwise an error code is returned.
2. Each APDU command is atomically updated, and one failed APDU will not affect other segments in the chain.
3. When in Plain Injection Mode, the read functionality for service data is forbidden. Instead it is recommended to use the PUT DATA command to write and read service data. Once the Plain Injection Mode changed to default mode, the service data could only be read by FREE READ SERVICE DATA.
4. When in Authentication Mode or Full Reset Mode, the command A71CL_External_Authentication must be executed before using the read functionality for service data.

4.2.1.4 Generate Key Pair Command

4.2.1.4.1 Definitions and scope

The Generate Key command is used to generate a complete asymmetric key public-private key pair. The specific key record is determined by the KID specified by data field. The public key value of the generated key pair is returned in the response message.

The generated key length is specified by the command data field, and the key length is in the range of 64 to 256 bytes and must be an integer multiple of 8. For example, RSA2048 is 256 bytes in length and 0x0800 in data field.

The KID of the public key must be consistent with that of the private key.

4.2.1.4.2 Pre-condition

The RSA key shall not exist.

In Default Mode, this command must establish secure channel before it executed.

In Authentication Mode or Full Reset Mode, this command's data field needs be wrapped by KEK key using RFC5649 first.

The KEK counter must be below its max try limit.

4.2.1.4.3 Post-condition

The lifecycle will transit to PERSONALIZED.

4.2.1.4.4 Command message

Generate Key Pair command message is shown.

Table 34. Generate Key Pair Command message

Code	Value
CLA	0x80
INS	0xF2
P1	See P1 parameter description
P2	0x00
Lc	0x04
Data	See definition
Le	0x00

Table 35. P1 parameter definition:

Value	Description
0x00	Generate key pair
0x01	Read public key value left when return data field is greater than 256 bytes

4.2.1.4.5 Command message data field

The command message data field contains the key type, the KID, and the two-byte key length which specifies the length of the asymmetric public-private key pair generated by the command.

Table 36. Data table

Definition	Byte number	Description
Key type	'01'	See Table 8 Key type description
Key KID	'01'	'00', KID of public key
Key length	'02'	The BIT length of the key

Note:

If the KID of the data field exists in the application, the generation of key pair fails and the error-reporting key exists.

4.2.1.4.6 Response message data field

The response message data field contains the generated public key mode.

The response message data field public key data format is shown in Table 37.

Table 37. Response Message Data Field

Type	Tag (T)	Length (L)	Value (V)	Tag (T)	Length (L)	Value (V)
RSA	6E	00 (256bytes)	Public key value N	65	04	'00010001'

4.2.1.4.7 Response message status code

Table 38. SecurityStorage – Status

SW1	SW2	Comment
0x69	0x82	Security is not satisfied
0x6A	0x81	Functionality is not supported
0x90	0x00	Executed correctly

See Secure Channel Error Condition for information on the additional warning status code from SCP that A71CL may send back.

4.2.1.5 AsymmetricCrypt Command

4.2.1.5.1 Definition and scope

AsymmetricCryptcommand is used for asymmetric algorithms using asymmetric operations, including encryption and decryption and signature calculation and verification .

4.2.1.5.2 Pre-condition

Secure message can be used in default mode.

A71CL must be in state PERSONALIZED".

4.2.1.5.3 Post-condition

-

4.2.1.5.4 Command message

Generate Key Pair command message is shown.

Table 39. Generate Key Pair Command message

Code	Value
CLA	0x80
INS	0xF2
P1	See P1 parameter description
P2	0x00
Lc	0x04
Data	See definition
Le	0x00

Table 40. P1 parameter definition:

Value	Description
0x00	Generate key pair
0x01	Read public key value left when return data field is greater than 256 bytes

4.2.1.5.5 Command message data field

The command message data field specifies the action to be done and the data to be processed. Table 41 shows the required content

Table 41. AsymmetricCrypt Command message data field

Definition	Number of bytes	Description	Remarks
Mode	1	0x51: encryption 0x52: decryption 0x53: signature 0x54: signature verification	When P1='00', partial data field exists
Algorithm type	1	00: RSA_NOPADDING 01: RSA_SHA1(RSA_PKCS1) 02: RSA_SHA256	
KID	1	KEY index, '00'	
Length	2	The to-be-processed data length	
Data	Length	See Table 42 for the data field of AsymmetricCrypt Command	

Table 42. Data Field For AsymmetricCrypt Command

P1	P2	Mode	Algorithm types	Data	Description
0x00	0x00/ 0x01	0x51: Encryption 0x52: Decryption	0x00: RSA NoPadding 0x01: RSA SHA1(PKCS1)	1-byte KID + 2 bytes length + data	P1 P2: 0001 indicates only one data block
		0x53:Signature 0x54:Verify signature	0x01: RSA SHA1(PKCS1) 0x02: RSA_SHA256		
0x01-0x20		——	——	Data	The nth data block
0x40	0x00	——	——	——	Le: xx more data to be returned

Note:

The signature verification data format shall be signature verification plaintext data + signature data concatenated without separator

4.2.1.5.6 Response message data field

Response message data field is an encrypted ciphertext and the Table 43 shows the required content:

Table 43. Response message data field

Definition	Requirements
Total response data length (2 bytes)	Return in the first response
RSA	Ciphertext length = algorithm mode length

4.2.1.5.7 Response message status code

Table 44. AsymmetricCrypt – Status

SW1	SW2	Comment
0x69	0x82	Security is not satisfied/ signature verification failed
0x6A	0x81	Functionality is not supported
0x90	0x00	Executed correctly

See Secure Channel Error Condition for more information on the error code.

4.2.1.6 SymmetricCrypt Command

4.2.1.6.1 Definition and scope

SymmetricCrypt command is used for symmetric encryption or MAC calculation.

4.2.1.6.2 Pre-condition

Secure message can be used in default mode.

When A71CL state is in state PERSONALIZED.

4.2.1.6.3 Post-condition

-

4.2.1.6.4 Command message

SymmetricCrypt message is shown.

Table 45. SymmetricCrypt Command message

Code	Value
CLA	0x80
INS	0xF6
P1	Block number: ‘00’~ ‘20’ Starting from ‘00’
P2	‘01’: last data block of to-be-processed data; ‘00’: not last data block of to-be-processed data
Lc	Length of command message data field

Code	Value
Data	Command message data
Le	The expected length of the response data

4.2.1.6.5 Command message data field

Command message data field contains to-be-processed data of the current block.

Table 46. Command message data field

Definition	Number of byte	Description
Mode	1	0x51: encryption 0x52: decryption 0x53: calculate MAC 0x54: verify MAC
Algorithm type	1	0x00: DES_CBC_NOPADDING 0x01: DES_ECB_NOPADDING 0x02: AES_CBC_NOPADDING 0x03: AES_ECB_NOPADDING 0x04: DES_CBC_ISO9797_M1 0x05: DES_CBC_ISO9797_M2 0x06: AES_CBC_ISO9797_M1 0x07: AES_CBC_ISO9797_M2
KID	1	KEY index, '00'.
Length	2	Total length of whole to-be-processed data over all blocks
Data	Length	To-be-processed data of this block, structure is described in Table 47 below

Note:

1.

The data can be split into several blocks and sent as cascaded commands. Only the first command message data field has algorithm pattern + algorithm types + KID + Length. The following message data fields only contain further data in the command message data field. The block number in P1 needs to count up for every further block with split data.

2.

By default, the symmetric algorithm's encryption and decryption padding option is set to No padding. All algorithm padding and calculation has to be completed outside of A71CL and is then given to the A71 to calculate data.

3.

MAC calculation of symmetric algorithm supports M1 and M2 Padding besides No padding mode.

4.

The structure of the data field is as following, the fields IV, Data and MAC have to be there with the length of "Number of bytes" dependend on the chosen "Mode" and "Algorithm type". There is no field separator, all bytes are concatenated.

Table 47. Data Field For SymmetricCrypt Command

Field	Mode	Algorithm types	Number of bytes	Description
IV	0x51: encryption 0x52: decryption 0x53: calculate MAC 0x54: verify MAC	0x00: DES_CBC_NOPADDING	8	Initial vector
		0x02: AES_CBC_NOPADDING 0x04: DES_CBC_ISO9797_M1 0x05: DES_CBC_ISO9797_M2 0x06: AES_CBC_ISO9797_M1 0x07: AES_CBC_ISO9797_M2	16	
	0x51: encryption 0x52: decryption	0x01: DES_ECB_NOPADDING 0x03: AES_ECB_NOPADDING	0	
Data	All	All		Data
MAC	0x54: verify MAC	0x00: DES_CBC_NOPADDING	8	MAC to be verified
		0x02: AES_CBC_NOPADDING 0x04: DES_CBC_ISO9797_M1 0x05: DES_CBC_ISO9797_M2 0x06: AES_CBC_ISO9797_M1 0x07: AES_CBC_ISO9797_M2	16	
	Others	All	0	

4.2.1.6.6 Response message data field

Depending on the selected mode the following response message data is returned:

Table 48. Response message data field

Mode	Number of byte	Description
'0x51': encryption '0x52': decryption	integral multiple of 0 or 8/16	Encryption/decryption operation is made and returned upon receipt of the key algorithm block length data
'0x53': calculate MAC	0 or 8/16	Return MAC value after receiving all the data and MAC value is the last block in the CBC algorithm.
'0x54': verify MAC	0	No data returned, only status

MAC algorithm refers to Section 6.1.

4.2.1.6.7 Response message status code

Table 49. SymmetricCrypt – Status

SW1	SW2	Comment
0x69	0x82	Security is not satisfied/MAC verification failed
0x6A	0x81	Functionality is not supported
0x90	0x00	Executed correctly

See Secure Channel Error Condition for more information on the error code.

4.2.1.7 GetID Command

4.2.1.7.1 Definition and scope

GetID command is used to read the ID value from security application.

4.2.1.7.2 Pre-condition

Secure message can be used in default mode.

ID value is stored.

4.2.1.7.3 Post-condition

-

4.2.1.7.4 Command message

GetID command message is shown

Table 50. GetID Command message

Code	Value
CLA	0x80
INS	0xF8
P1	0x00
P2	0x00
Lc	Does not exist
Data	Does not exist
Le	'XX'

4.2.1.7.5 Command message data field

Command message data field does not exist.

4.2.1.7.6 Response message data field

The requirements for the response message data field are as follows:

The returned data is the OEM's information formatted as follows:

Table 51. Response message data field

Information	Number of byte	Description
OEM identification	'02'	'8182' = NXP Semiconductors
Length	'01'	ID length
Device ID	Length	ID string

4.2.1.7.7 Response message status code

Table 52. GetID – Status

SW1	SW2	Comment
0x6A	0x80	Wrong data

SW1	SW2	Comment
0x6A	0x81	Functionality is not supported
0x90	0x00	Executed correctly

See Secure Channel Error Condition for more information on the error code.

4.2.1.8 GetVendorInfo Command

4.2.1.8.1 Definition and scope

The Get Vendor Info command is used to request information of OEM from the A71CL applet.

4.2.1.8.2 Pre-condition

Secure message can be used in default mode.

4.2.1.8.3 Post-condition

-

4.2.1.8.4 Command message

The Get Vendor Info command message is shown

Table 53. GetVendorInfo Command message

Code	Value
CLA	0x80
INS	0xFC
P1	0x00
P2	0x00
Lc	Does not exist
Data	Does not exist
Le	'XX'

4.2.1.8.5 Command message data field

Command message data field does not exist.

4.2.1.8.6 Response message data field

The returned data is the OEM's information formatted as follows:

Table 54. Response message data

Information	Length
OEM identification	2 bytes = '8182' - NXP Semiconductors
Version information	8 bytes
Device configuration options	4 bytes
Available space	2 bytes

Information	Length
Extension bit	4 bytes

4.2.1.8.7 Response message status code

Table 55. GetVendorInfo – Status

SW1	SW2	Comment
0x69	0x84	Data is invalid
0x90	0x00	Executed correctly

See Secure Channel Error Condition for more information on the error code.

4.2.1.9 Disable “Plain Injection Mode” Command

4.2.1.9.1 Definition and scope

This command is activated once PLAIN INJECTION MODE is configured during installation. When this command is executed, all functionality of plain injection mode is disabled, including the Disable Plain Injection Mode and Put Data commands.

4.2.1.9.2 Security condition:

None[8].

4.2.1.9.3 Pre-condition

A71CL must be in state PERSONALIZED”.

4.2.1.9.4 Post-condition

After successful execution, this command is disabled.

4.2.1.9.5 Command message

Table 56. Disable “Plain Injection Mode” Command message

Code	Value
CLA	0x80
INS	0xFE
P1	0x00
P2	0x00
Lc	0x00
Data	Not present

4.2.1.9.6 Command message data field

Command message data field does not exist.

8 Command shall send without SCP or KEK wrapping since command message data field does not exist.

4.2.1.9.7 Response message data field

Response message data field does not exist.

4.2.1.9.8 Response message status code

Table 57. Disable "Plain Injection MODE" – Status

SW1	SW2	Comment
0x90	0x00	OK
0x6D	0x00	Command is disabled

4.2.1.10 Put Data Command

4.2.1.10.1 Definition and scope

The Put Data is used to either:

- Replace an existing KEK with a new KEK when authentication mode is set or Full reset Mode configuration.
- Add/Update OEM service data.

Once the KEK has been set, data field may only be sent wrapped by KEK using RFC5649 from off-card. Plaintext transmission of the data field is not allowed. The other key is not allowed to add through this command.

Note:

1.When writing service data, the maximum record length should not exceed 255 bytes.

2.The service data is searched by record number which does not need update in chain.

3. each APDU command is atomically updated, and one failed APDU should not affect other segments in the chain.

4.2.1.10.2 Security condition:

-

4.2.1.10.3 Pre-condition

To replace KEK, mode configuration must be set to Full reset Mode or Authentication Mode.

To add/update OEM service data, mode configuration must be set to or Authentication Mode or Plain Injection Mode.

Once KEK is set, service data must be wrapped by RFC5649 using KEK. The KEK counter must be below its max try limit.

The failed of RFC5649 unwrapping checking will increase the KEK counter, once the KEK counter meet the max try limit counter, the KEK will be locked. The KEK counter is restored when RFC5649 unwrapping passes.

4.2.1.10.4 Post-condition

-

4.2.1.10.5 Command message

Table 58. Put Data Command message

Code	Value
CLA	0x84
INS	0x12
P1	See P1 parameter description
P2	0x00
Lc	Length of the data filed
Data	RFC5649 wrapped data or plaintext[1]

[1] Once in PLAIN INJECTION MODE, only OEM service data with plaintext is supported.

Table 59. Definition of most significant bit “Bit 7” of P1

Bit7	Description
1	KEK key instruction
0	OEM service data instruction

Table 60. Definition of most significant bit “Bit 6” of P1 in the condition of “Bit7”

Bit7	Bit6	Description
1: KEK key instruction	1	The last data to be processed
	0	RFU
0: OEM service data instruction	x	Put Service Data

Table 61. Definition of most significant bit “Bit5~Bit0” of P1:

Bit5~Bit0	Description
‘XX’	Value range of the block number of KEK key or the record number of cascaded OEM service data: "00"~"20"

4.2.1.10.6 Command message data field

Service data or KEK data.

4.2.1.10.7 Response message data field

Response message data field does not exist.

4.2.1.10.8 Response message status code

Table 62. Disable “Plain Injection MODE” – Status

SW1	SW2	Comment
‘69’	‘82’	Security status not satisfied

SW1	SW2	Comment
'69'	'83'	KEK is blocked
'69'	'85'	Conditions not satisfied
'6A'	'88'	KEK not found
'90'	'00'	Successfully processed

4.2.1.11 Free Read Service Data

4.2.1.11.1 Definition and scope

The Free Read Service Data is used to read the service data written through PUT DATA. The Plain text return from this command can be freely read by off-card entity without authentication.

4.2.1.11.2 Security condition:

None.

4.2.1.11.3 Pre-condition

A71CL must be in state PERSONALIZED"

4.2.1.11.4 Post-condition

-

4.2.1.11.5 Command message

Table 63. Free Read Service Data command message

Code	Value
CLA	0x80
INS	0x71
P1	Record Number with high byte
P2	Record Number with low byte
Le	0x00

4.2.1.11.6 Response message data field

Table 64. Response message data field table

Value	Comment
Plaintext data chunk	Byte array contains service data with plaintext controlled by record number.

4.2.1.11.7 Response message status code

Table 65. Response message status code

SW1	SW2	Comment
0x67	0x00	Length is incorrect
0x6A	0x82	Record not found

SW1	SW2	Comment
0x90	0x00	No error

4.2.1.12 A71CL_External_Authentication

4.2.1.12.1 Definition and scope

A71CL_External_Authentication is used to verify the off-card entity. 16 bytes Challenge are needed in this session.

The A71CL applet will decrypt cryptogram from an off-card entity using KEK and then compare with the challenge number from GET CHALLENGE. If the compared numbers don't match, the KEK counter is decreased. The max try limit is 3: after three failed attempts the KEK is locked forever. The KEK counter is restored when an authentication passes.

4.2.1.12.2 Security condition:

-

4.2.1.12.3 Pre-condition

Mode configuration MUST be set to Full reset Mode or Authentication Mode.

Data Field of the command shall be encrypted by KEK.

KEK counter must be below its max try limit.

4.2.1.12.4 Post-condition

-

4.2.1.12.5 Command message

Table 66. A71CL_External_Authentication command message

Code	Value
CLA	0x00
INS	0x82
P1	0x00
P2	0x00
Lc	0x10
Data	16 bytes cryptogram

4.2.1.12.6 Command message data field

16 bytes cryptogram which is calculated with 16-bytes challenges retrieved from Get Challenge command using AES_ECB by KEK.

4.2.1.12.7 Response message data field

Response message data field does not exist.

4.2.1.12.8 Response message status code

Table 67. Response message status code

SW1	SW2	Comment
0x69	0x82	Security status not satisfied
0x69	0x85	Conditions not satisfied
0X90	0x00	Successfully processed

4.2.1.13 Full Reset

4.2.1.13.1 Definition and scope

The Full Reset command can delete all the user data as well as key value. The key object will be set to uninitialized.

A successful execution of the A71CL_EXTERNAL_AUTHENTICATION command shall precede this command.

4.2.1.13.2 Security condition:

None.

4.2.1.13.3 Pre-condition

Mode configuration must be set to Full reset Mode.

A71CL_External_Authentication must be executed successfully.

4.2.1.13.4 Post-condition

All user data and key are deleted except the KEK remains the same. The A71CL applet returns to the state UNPERSONALIZED.

4.2.1.13.5 Command message

Table 68. Full Reset command message

Code	Value
CLA	0x80
INS	0xEA
P1	0x00
P2	0x00
Lc	0x00
Data	Not present

4.2.1.13.6 Command message data field

Command message data field does not exist.

4.2.1.13.7 Response message data field

Response message data field does not exist.

4.2.1.13.8 Response message status code

Table 69. Response message status code

SW1	SW2	Comment
0x69	0x82	Security status not satisfied
0x69	0x83	KEK is blocked
0x69	0x85	Conditions not satisfied
0x6A	0x88	KEK not found
0x6C	0xCy	y tries remain (eg.2=6CC2,1=6CC1)
0X90	0x00	Successfully processed

5 A71CL Implementation Notes

5.1 Hardware Interface

Table 70 lists the protocols in use.

Table 70. Hardware interface specifications

HW interface	Protocol specification	Protocol details
I2C	SCI2C	an195015 - Application note SCIIC Protocol Specification (1.5)

5.2 EEPROM Write Access

The hardware supports a limited number of EEPROM write accesses. Integrators have to take care to avoid unnecessary EEPROM writes due to calling commands that access & write to EEPROM. Table 71 gives an overview of commands that write to EEPROM.

Table 71. Commands writing to EEPROM

Command	Remark
Disable Plain Injection Mode	
Put Data	
A71CL_External_ Authentication	
Full Reset	
SecurityStorageData	
GenerateKeyPair	

6 Annex A (Normative) CMAC Algorithm

6.1 A.1 CMAC Algorithm

CMAC algorithm adopts “MAC algorithm 1” as specified in ISO 9797-1.

7 Annex B (Normative) A71CL configuration Information

7.1 Configuration Information For BaiDu Cloud

7.1.1 Provisioning Mode Information For Baidu Cloud

The Configuration “800100AC0100” is appied as specified in 2.1 Provisioning Mode Configuration which stands for “RSA CRT is enabled and the Default Mode is set”.

7.1.2 Provisioning Key Type Information For Baidu Cloud

One RSA CRT private key and one Baidu Cloud public key are provisioned.

7.1.3 AID Information For Baidu Cloud

- Package AID is defined as follows:

‘com.nxp.IoT.A7’:

0x63,0x6F,0x6D,0x2E,0x6E,0x78,0x70,0x2E,0x49,0x6F,0x54,0x2E,0x41,0x37

- Module ID is defined as follows

‘com.nxp.IoT.app’:

0x63,0x6F,0x6D,0x2E,0x6E,0x78,0x70,0x2E,0x49,0x6F,0x54,0x2E,0x61,0x70,0x70

- Instance AID is defined as follows:

0xA0,0x00,0x00,0x42,0x61,0x69,0x64,0x75,0x59,0x75,0x6E,0x2E,0x49,0x6F,0x54

7.1.4 Device ID Information For Baidu Cloud

The Device ID for Baidu is a 16-byte-long string which is provided from Baidu.

8 Document Management

8.1 Abbreviations and Terminology

Table 72. Abbreviations

Abbreviation	Description
AID	Application Identifier
APDU	Application Protocol Data Unit
ATR	Answer to Reset
b	Binary
BER	Basic Encoding Rules
BWI	Block Waiting Time Integer
CLA	Class Byte of the Command Message
CWI	Character Waiting Time Integer
DEA	Data Encryption Algorithm
DES	Data Encryption Standard
DF	Dedicated File
EDC	Error Detection Code
EF	Elementary File
Etu	Elementary Time Unit
FCI	File Control Information
FID	File Identifier
GND	Ground
Hex.	Hexadecimal
IC	Integrated Circuit
ICC	Integrated Circuit Card
IEC	International Electrotechnical Commission
INS	Instruction Byte of Command Message
ISO	International Standardization Organization
Lc	the actual length of the command data field sent by terminal
Le	the maximum expected length of the response data
LEN	Length
MAC	Message Authentication Code
MF	Master File
P1	Parameter 1
P2	Parameter 2
PBOC	People's Bank of China
PIN	Personal Identification Number

Abbreviation	Description
PIX	Proprietary Application Identifier Extension
PSA	Payment System Application
PSAM	Purchase Secure Access Module
PSE	Payment System Environment
RFU	Reserved for Future Use
RID	Registered Application Provider Identify
RSA	Rivest,Shamir,Adleman
RST	Reset
SAM	Secure Access Module
SFI	Short File Identifier
SHA	Secure Hash Algorithm
SW1	Status Word One
SW2	Status Word Two

8.2 Referenced Documents

Table 73. Referenced documents

Doc ID	Doc Title
[RFC5649]	Advanced Encryption Standard (AES) Key Wrap with Padding Algorithm [August 2009]
[RFC3394]	Advanced Encryption Standard (AES) Key Wrap Algorithm [September 2002]
[ISO/IEC 7816-4:2013]	Identification cards -- Integrated circuit cards -- Part 4: Organization, security and commands for interchange [2013-04-15]

Revision history

Revision history

Rev	Date	Description
v.1.0	20181214	Initial version

9 Legal information

9.1 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

9.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors. In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory. Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification. Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products. NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer. In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages. Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

9.3 Licenses

ICs with DPA Countermeasures functionality

LICENSED DPA™ COUNTERMEASURES

NXP ICs containing functionality implementing countermeasures to Differential Power Analysis and Simple Power Analysis are produced and sold under applicable license from Cryptography Research, Inc.

9.4 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

Kinetis — is a trademark of NXP B.V.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile — are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

Tables

Figures

Contents

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 14 December 2018
Document identifier: AN12305
Document number: 515410