

Smartcard API	1
.....	1
BS2CSNCard	1
BS2SmartCardHeader	2
BS2SmartCardCredentials	4
BS2AccessOnCardData	4
BS2SmartCardData	5
BS2Card	5

Smartcard API

API

- BS2_ScanCard:
- BS2_WriteCard: SmartCard
- BS2_EraseCard: SmartCard

BS2CSNCard

```
typedef struct {
    uint8_t type;
    uint8_t size;
    uint8_t data[BS2_CARD_DATA_SIZE];
} BS2CSNCard;
```

1. *type*

Access

가

0x00		
0x01	CSN	
0x02	Secure	
0x03	Access	
0x06	QR	
0x0A	Wiegand	BS2WiegandConfig.format (BS2WiegandConfig.CSNIndex BS2WiegandConfig.CardMask7} 0 .
0x0B	Config	
0x1A	Wiegand	BS2WiegandMultiConfig.formats[0]
0x2A	Wiegand	BS2WiegandMultiConfig.formats[1]
0x3A	Wiegand	BS2WiegandMultiConfig.formats[2]
0x4A	Wiegand	BS2WiegandMultiConfig.formats[3]
0x5A	Wiegand	BS2WiegandMultiConfig.formats[4]
0x6A	Wiegand	BS2WiegandMultiConfig.formats[5]
0x7A	Wiegand	BS2WiegandMultiConfig.formats[6]
0x8A	Wiegand	BS2WiegandMultiConfig.formats[7]
0x9A	Wiegand	BS2WiegandMultiConfig.formats[8]
0xAA	Wiegand	BS2WiegandMultiConfig.formats[9]
0xBA	Wiegand	BS2WiegandMultiConfig.formats[10]
0xCA	Wiegand	BS2WiegandMultiConfig.formats[11]
0xDA	Wiegand	BS2WiegandMultiConfig.formats[12]

0xEA	Wiegand	BS2WiegandMultiConfig.formats[13]
0xFA	Wiegand	BS2WiegandMultiConfig.formats[14]

2. size

3. data

BS2SmartCardHeader

```
typedef struct {
    uint16_t hdrCRC;
    uint16_t cardCRC;
    BS2_CARD_TYPE cardType;
    uint8_t numOfTemplate;
    uint16_t templateSize;
    uint16_t issueCount;
    uint8_t duressMask;
    uint8_t cardAuthMode;
    uint8_t useAlphanumericID;
    uint8_t cardAuthModeEx;
    uint8_t numOfFaceTemplate;
    uint8_t reserved[1];
} BS2SmartCardHeader;
```

1. **hdrCRC**
card header . (cardCRC - reserved)
2. **cardCRC**
card data . (BS2SmartCardHeader.cardType - BS2SmartCardData.accessOnData)
3. **cardType**

0x00	
0x01	CSN
0x02	Secure
0x03	Access
0x0A	Wiegand
0x0B	Config

4. numOfTemplate

AOC	template	BS2SmartCardCredentials
-----	----------	---

AOC	template	, numOfFacetemplate	0
-----	----------	---------------------	---

5. *templateSize*

BioStar 2

6. *issueCount*

7. duressMask

8. *cardAuthMode*

FaceStation F2

FaceStation F2 **cardAuthModeEx**

2	
3	
4	PIN
5	PIN
6	, , PIN
254	
255	()

9. *useAlphanumericID*

Alphanumeric ID	flag
-----------------	------

10. *cardAuthModeEx*

FaceStation F2 **cardAuthMode**

[+ V2.7.1] FaceStation F2

	1	2	3
21			
22			
23			
24		PIN	
25			
26		PIN	
27		PIN	
28		PIN	
29			
30			PIN
31			
32			PIN

	1	2	3
33			PIN
34			PIN
35			PIN
254			
255	(		

11. numOfFaceTemplate

template 가 (:384, :552)
BS2SmartCardCredentials templateData

AOC template BS2SmartCardCredentials

AOC template , numOfTemplate 0

12. reserved

BS2SmartCardCredentials

```
typedef struct {
    uint8_t pin[BS2_PIN_HASH_SIZE];
    uint8_t templateData[BS2_SMART_CARD_MAX_TEMPLATE_COUNT *
BS2_FINGER_TEMPLATE_SIZE];
} BS2SmartCardCredentials;
```

1. pin
pin code

2. templateData
4

BS2AccessOnCardData

```
typedef struct {
    uint16_t accessGroupID[BS2_SMART_CARD_MAX_ACCESS_GROUP_COUNT];
    BS2_DATETIME startTime;
    BS2_DATETIME endTime;
```

```
} BS2AccessOnCardData;
```

1. *accessGroupID*

2. *startTime*

가 , 0

3. *endTime*

가 , 0

BS2SmartCardData

```
typedef struct {
    BS2SmartCardHeader header;
    uint8_t cardID[BS2_CARD_DATA_SIZE];
    BS2SmartCardCredentials credentials;
    BS2AccessOnCardData accessOnData;
} BS2SmartCardData;
```

1. *header*

SmartCard

2. *cardID*

Access on Card 32 byte Card ID ,
 Secure Credential card 24 byte Card ID
 Secure Credential Card(SCC) , 32 byte Card ID(24 bytes), issueCount(4 byte) Time
 Stamp(4 byte)
 , BS2UserBlob cardObjs array가 SC Card , SC Card cardObjs

3. *credentials*

pin code

4. *accessOnData*

AOC 가

BS2Card

```
typedef struct {
    uint8_t isSmartCard;
    union {
        BS2CSNCard card;
        BS2SmartCardData smartCard;
    };
} BS2Card;
```

1. *isSmartCard*

SmartCard flag

2. *card*

3. *smartCard*

SmartCard

From:

<https://kb.supremainc.com/kbtest/> - **BioStar 2 Device SDK**

Permanent link:

https://kb.supremainc.com/kbtest/doku.php?id=ko:smartcard_api

Last update: **2021/09/14 14:37**